US011838271B2

(12) United States Patent
Foxhoven et al.

(10) Patent No.: US 11,838,271 B2
(45) Date of Patent: Dec. 5, 2023

(54) **PROVIDING USERS SECURE ACCESS TO BUSINESS-TO-BUSINESS (B2B) APPLICATIONS**

(71) Applicant: **Zscaler, Inc.**, San Jose, CA (US)

(72) Inventors: **Patrick Foxhoven**, San Jose, CA (US); **John A. Chanak**, Saratoga, CA (US); **William Fehring**, Sunnyvale, CA (US); **Manoj Apte**, San Jose, CA (US); **Kunal Shah**, San Jose, CA (US); **Dhawal Sharma**, San Jose, CA (US)

(73) Assignee: **Zscaler, Inc.**, San Jose, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 14 days.

(21) Appl. No.: **17/084,704**

(22) Filed: **Oct. 30, 2020**

(65) **Prior Publication Data**

US 2021/0136041 A1 May 6, 2021

Related U.S. Application Data

(63) Continuation-in-part of application No. 16/800,307, filed on Feb. 25, 2020, which is a continuation of
(Continued)

(51) **Int. Cl.**
H04L 9/40 (2022.01)
G06F 9/54 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 63/0272*** (2013.01); ***G06F 9/547*** (2013.01); ***H04L 9/006*** (2013.01);
(Continued)

(58) **Field of Classification Search**
CPC ... H04L 63/0272; H04L 9/006; H04L 9/0894; H04L 9/14; H04L 9/30; H04L 9/3263;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,636,923 B1 10/2003 Meirsman et al.
8,869,259 B1 10/2014 Udupa et al.
(Continued)

OTHER PUBLICATIONS

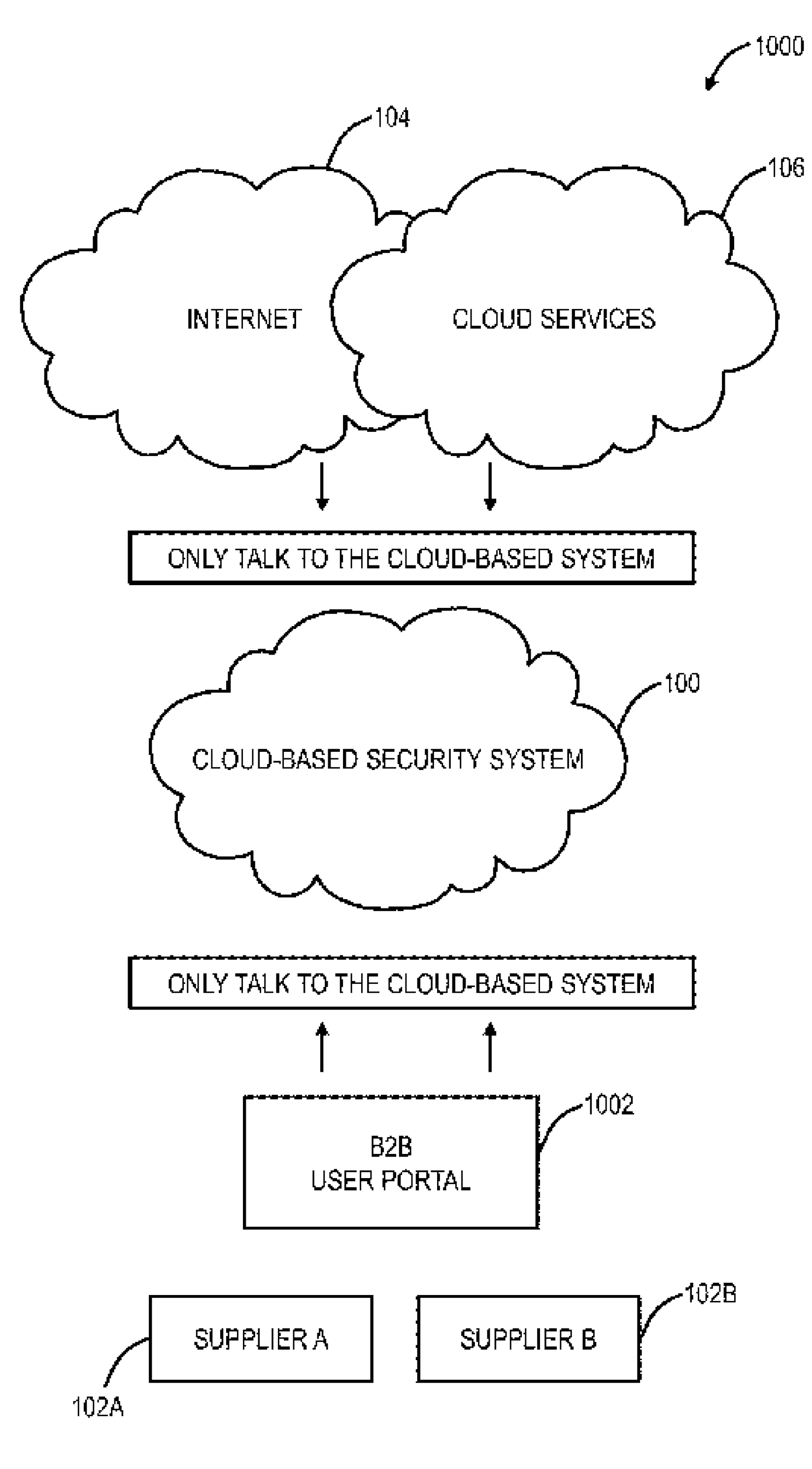
J. R. Vic Winkler, "Securing the Cloud: Cloud Computer Security Techniques and Tactics", May 2011, Syngress Publishing, Full Text.
(Continued)

Primary Examiner — Bryan F Wright
(74) *Attorney, Agent, or Firm* — Baratta Law PLLC; Lawrence A. Baratta, Jr.; Ryan Odessa

(57) **ABSTRACT**

Systems and methods include, responsive to a request from a user for one or more Business-to-Business (B2B) applications, redirecting the request, by a cloud-based system, to an identity provider to authorize the user; displaying the one or more B2B applications that the user is authorized to access; responsive to a selection of a B2B application of the one or more B2B applications, creating a first tunnel from the B2B application to the cloud-based system; and stitching the first tunnel between the B2B application and the cloud-based system with a second tunnel between the user and the cloud-based system. The systems and methods further include, responsive to the user being unauthorized for any of the one or more B2B applications, omitting the one or more B2B applications from the displaying, such that the one or more B2B applications are invisible to the user.

19 Claims, 22 Drawing Sheets



Related U.S. Application Data

application No. 15/986,874, filed on May 23, 2018, now Pat. No. 10,616,180, which is a continuation-in-part of application No. 15/158,153, filed on May 18, 2016, now Pat. No. 10,375,024.

(51) **Int. Cl.**

H04L 9/14	(2006.01)
H04L 9/32	(2006.01)
H04L 9/08	(2006.01)
H04L 9/00	(2022.01)
H04L 67/1021	(2022.01)
H04L 9/30	(2006.01)
H04L 67/01	(2022.01)
H04L 61/59	(2022.01)
H04L 61/4511	(2022.01)

(52) **U.S. Cl.**

CPC ***H04L 9/0894*** (2013.01); ***H04L 9/14*** (2013.01); ***H04L 9/30*** (2013.01); ***H04L 9/3263*** (2013.01); ***H04L 63/029*** (2013.01); ***H04L 63/0823*** (2013.01); ***H04L 63/0876*** (2013.01); ***H04L 67/01*** (2022.05); ***H04L 67/1021*** (2013.01); *H04L 61/4511* (2022.05); *H04L 61/59* (2022.05)

(58) **Field of Classification Search**

CPC .. H04L 63/0823; H04L 63/0876; H04L 67/42

USPC 713/156

See application file for complete search history.

(56) **References Cited**

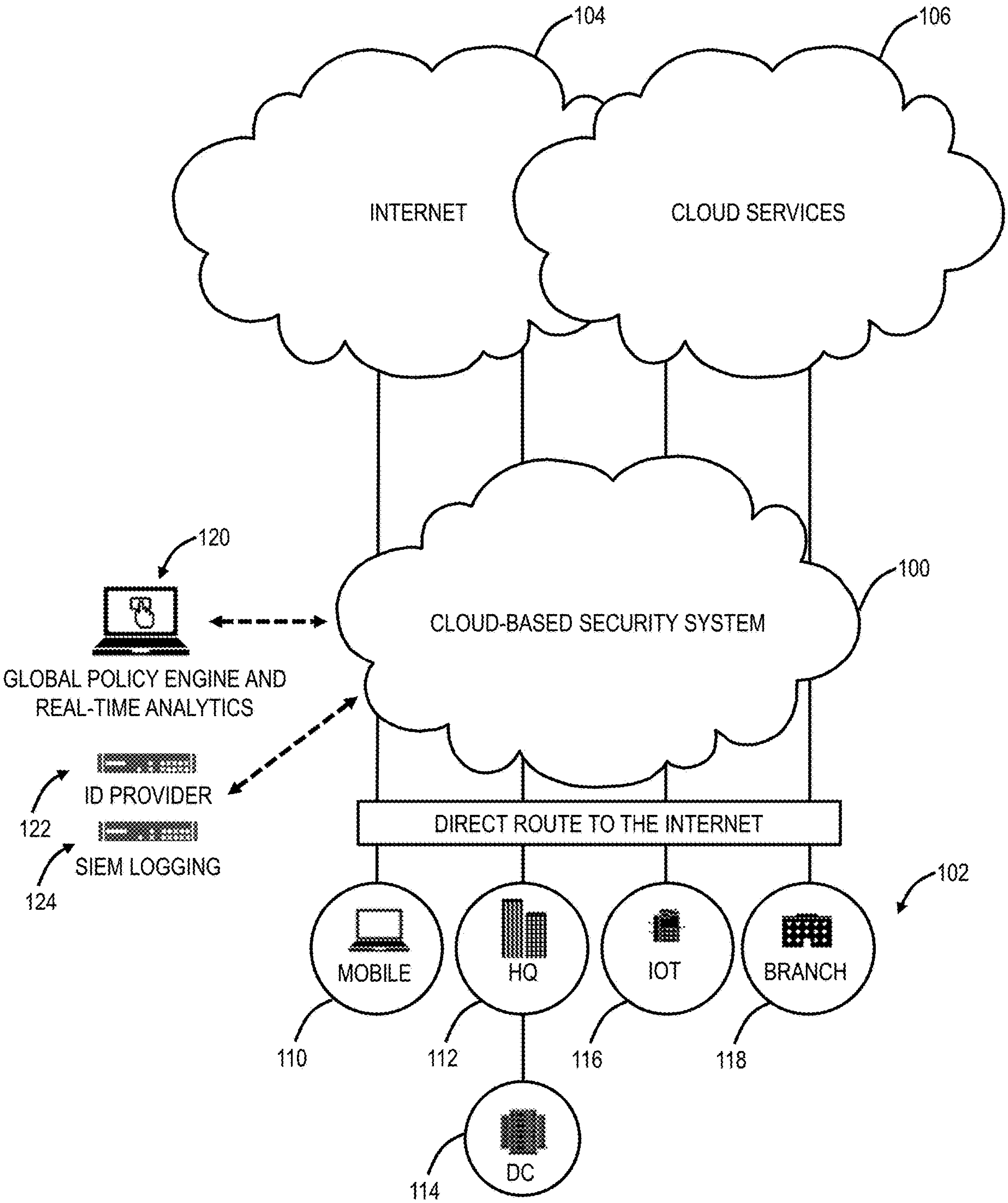
U.S. PATENT DOCUMENTS

2002/0032763	A1*	3/2002	Cox	H04L 69/329 709/223
2002/0111818	A1*	8/2002	Morar	G06Q 20/40 705/26.1
2006/0074618	A1	4/2006	Miller et al.	
2007/0042756	A1	2/2007	Perfetto et al.	
2007/0214232	A1	9/2007	Belimpasakis et al.	
2008/0004886	A1*	1/2008	Hames	G06Q 30/0645 705/26.1
2008/0307519	A1	12/2008	Curcio	
2009/0129271	A1	5/2009	Ramankutty et al.	
2011/0231840	A1*	9/2011	Burch	G06F 9/5077 718/1
2011/0296486	A1	12/2011	Burch et al.	
2011/0310899	A1	12/2011	Alkhatib et al.	
2012/0023325	A1	1/2012	Lai	
2012/0144034	A1	6/2012	Mccarty	
2012/0158821	A1*	6/2012	Barros	G06F 9/5055 707/769
2012/0185913	A1	7/2012	Martinez et al.	
2012/0216244	A1*	8/2012	Kumar	G06F 21/57 726/1
2012/0246475	A1	9/2012	Yoo et al.	
2012/0266231	A1*	10/2012	Spiers	H04L 9/3234 726/12
2012/0281708	A1	11/2012	Chauhan et al.	
2013/0347072	A1	12/2013	Dinha	
2014/0006347	A1*	1/2014	Qureshi	H04L 9/0822 707/621
2014/0022586	A1	1/2014	Zehler	
2014/0052837	A1*	2/2014	Prasad	H04L 41/5051 709/223
2014/0259093	A1	9/2014	Narayanaswamy et al.	
2014/0282817	A1	9/2014	Singer et al.	
2014/0354453	A1*	12/2014	Fourie	G08G 1/144 340/932.2
2015/0372982	A1	12/2015	Herle et al.	
2016/0057211	A1*	2/2016	Thapliyal	H04L 67/02 709/203

OTHER PUBLICATIONS

Stephen R. Smoot, "Private Cloud Computing: Consolidation, Virtualization, and Service-Oriented Infrastructure", Oct. 2011, Morgan Kaufman Publishers, Inc. Full Text.

* cited by examiner



104
106
INTERNET
CLOUD SERVICES
120
100
CLOUD-BASED SECURITY SYSTEM
GLOBAL POLICY ENGINE AND REAL-TIME ANALYTICS
ID PROVIDER
122
SIEM LOGGING
124
DIRECT ROUTE TO THE INTERNET
102
MOBILE
HQ
IOT
BRANCH
110
112
116
118
DC
114

FIG. 1

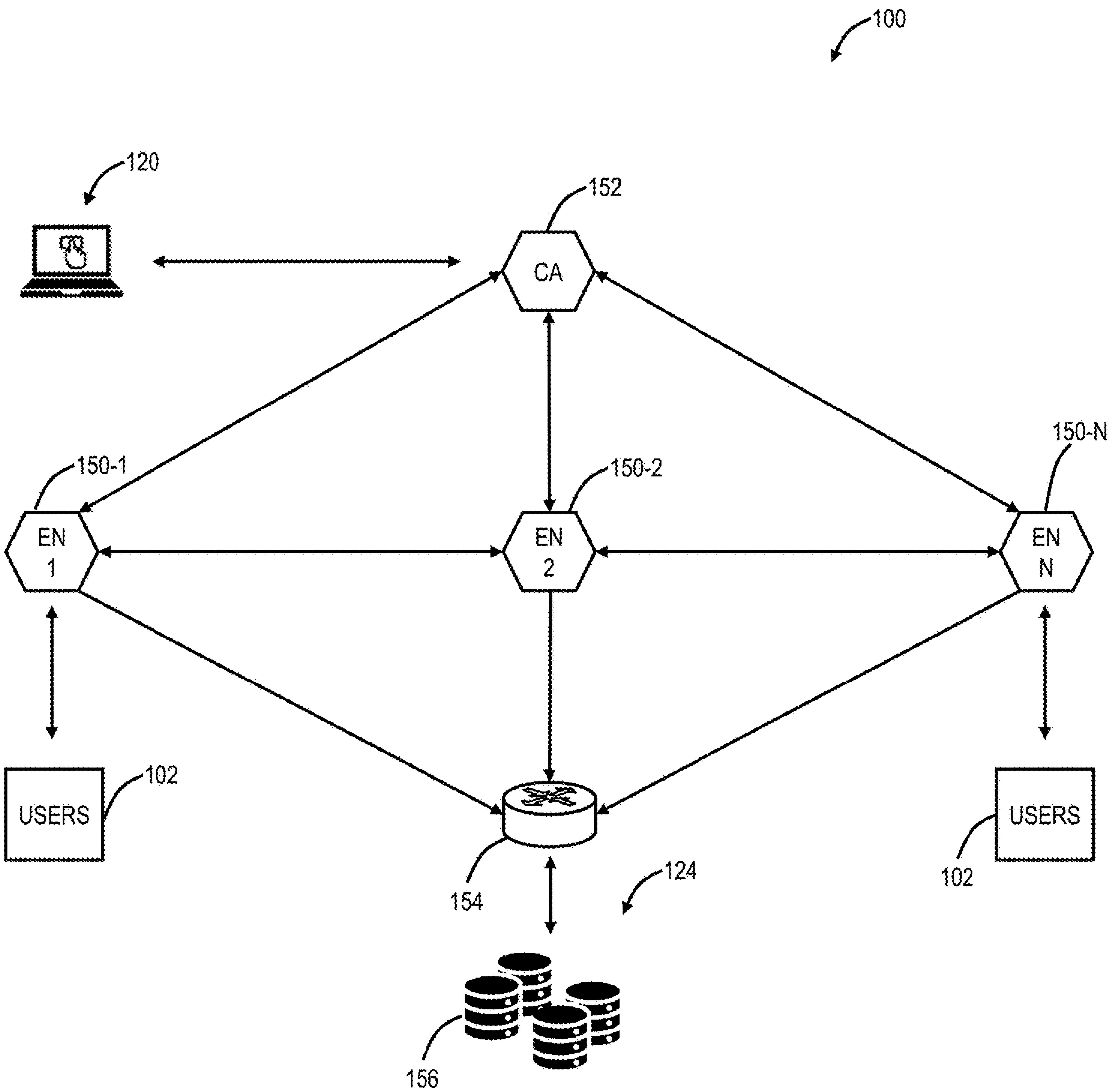

100
120
152
CA
150-1
150-2
150-N
EN
1
EN
2
EN
N
USERS
102
USERS
102
154
124
156

FIG. 2

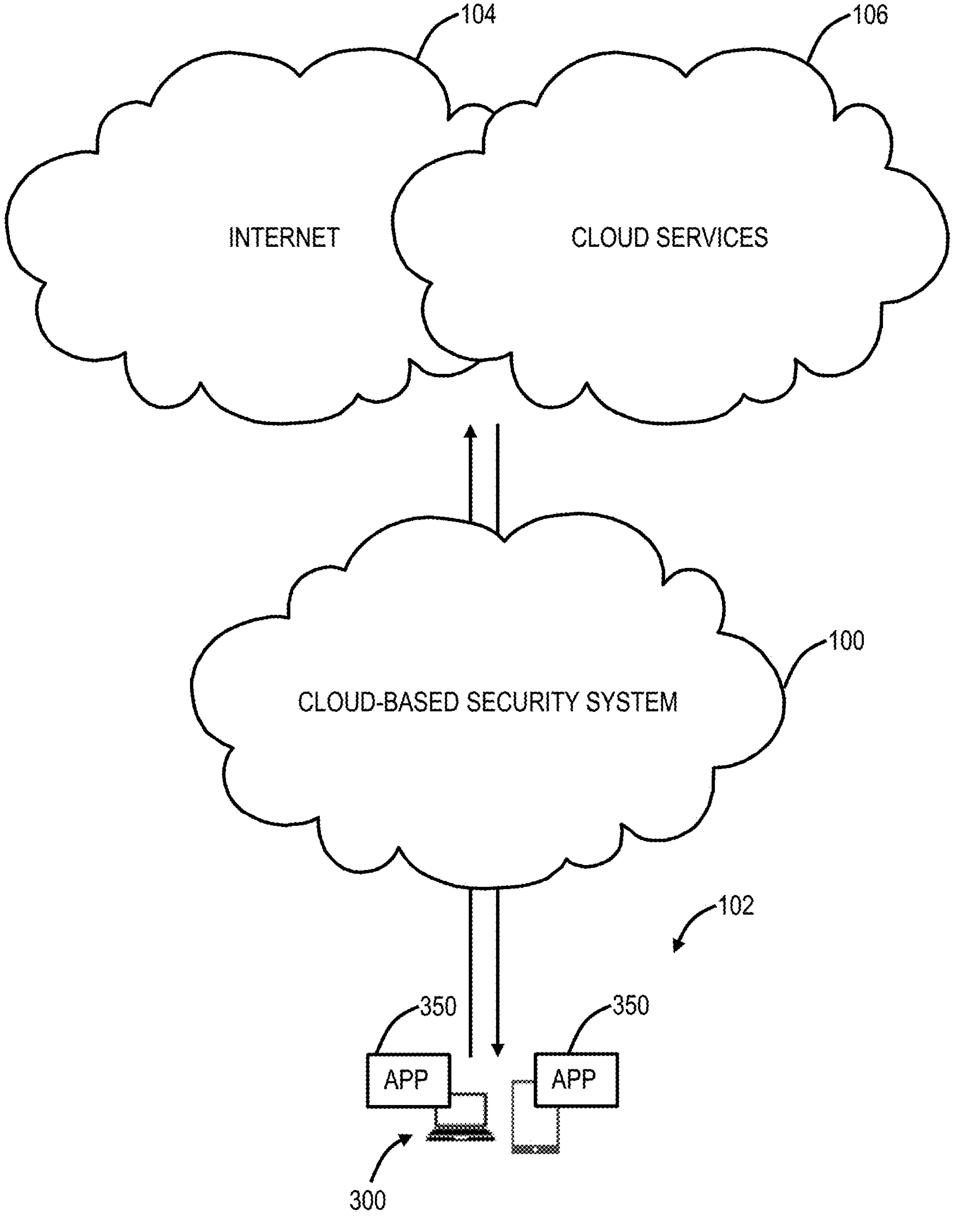

104
106
INTERNET
CLOUD SERVICES
100
CLOUD-BASED SECURITY SYSTEM
102
350
350
APP
APP
300

FIG. 3

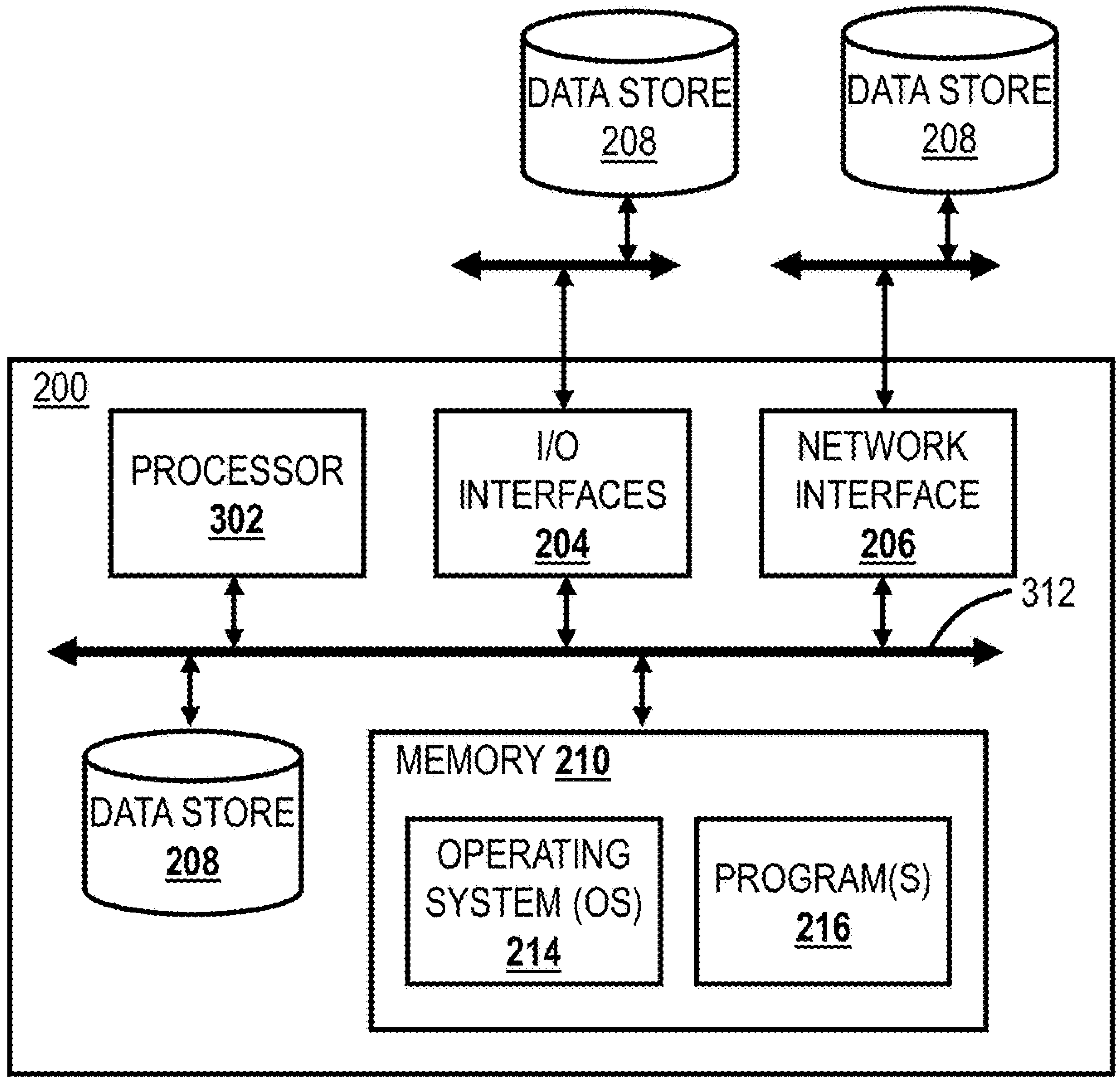

DATA STORE
208
DATA STORE
208
200
PROCESSOR
302
I/O
INTERFACES
204
NETWORK
INTERFACE
206
312
DATA STORE
208
MEMORY 210
OPERATING
SYSTEM (OS)
214
PROGRAM(S)
216

FIG. 4

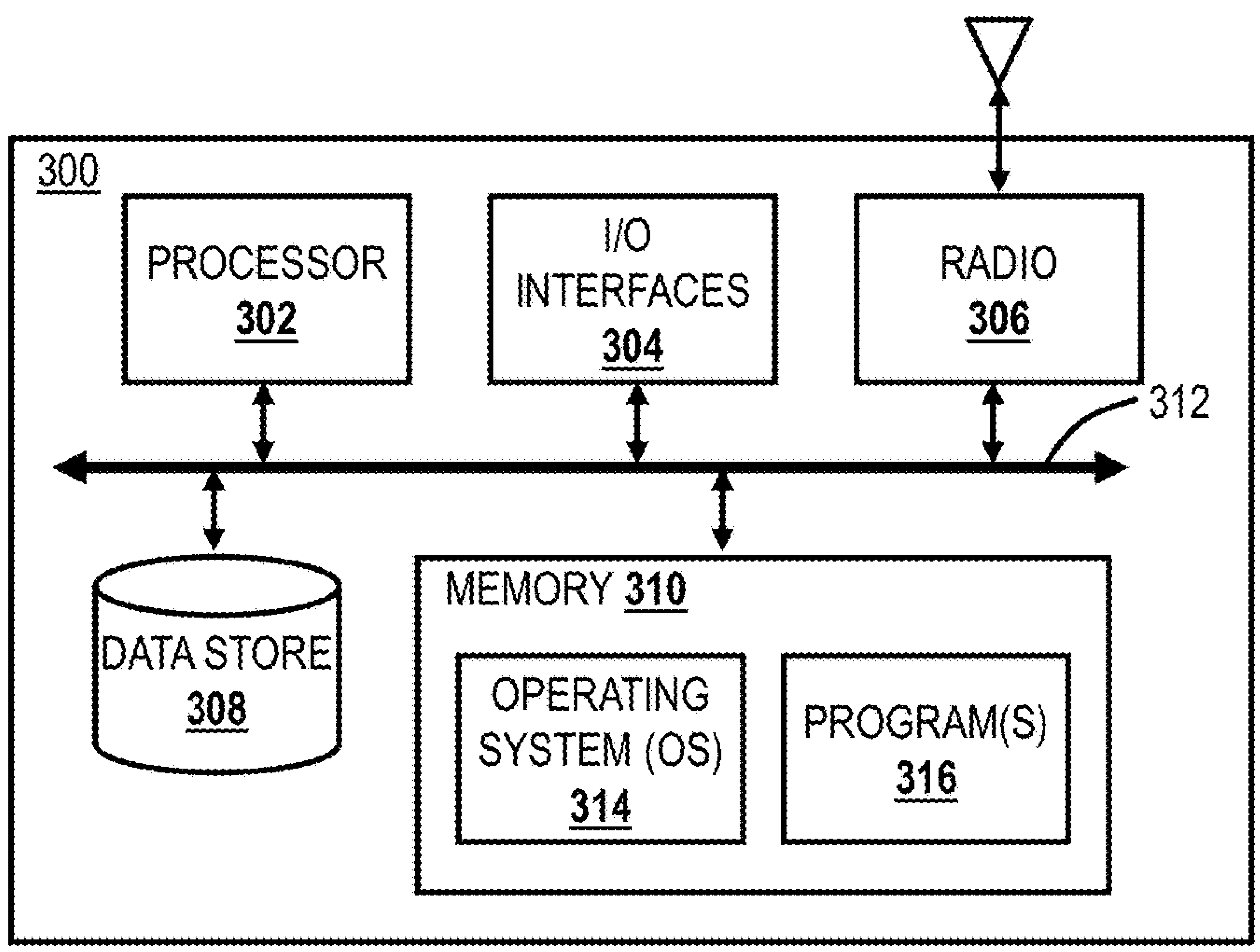

300
PROCESSOR
302
I/O
INTERFACES
304
RADIO
306
312
DATA STORE
308
MEMORY 310
OPERATING
SYSTEM (OS)
314
PROGRAM(S)
316

FIG. 5

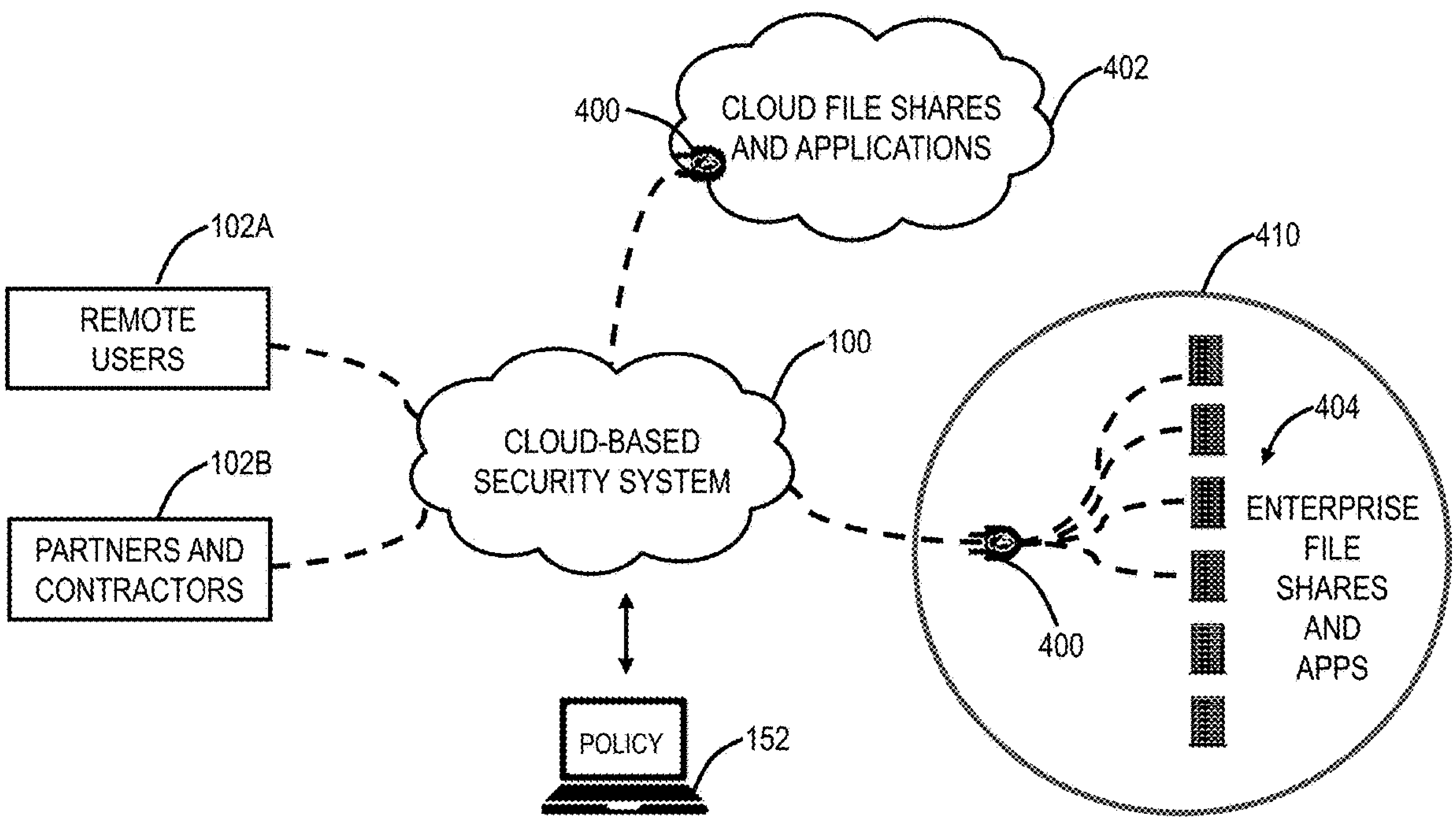

CLOUD FILE SHARES AND APPLICATIONS
402
400
102A
REMOTE USERS
410
100
404
CLOUD-BASED SECURITY SYSTEM
102B
PARTNERS AND CONTRACTORS
ENTERPRISE FILE SHARES AND APPS
400
POLICY
152

FIG. 6

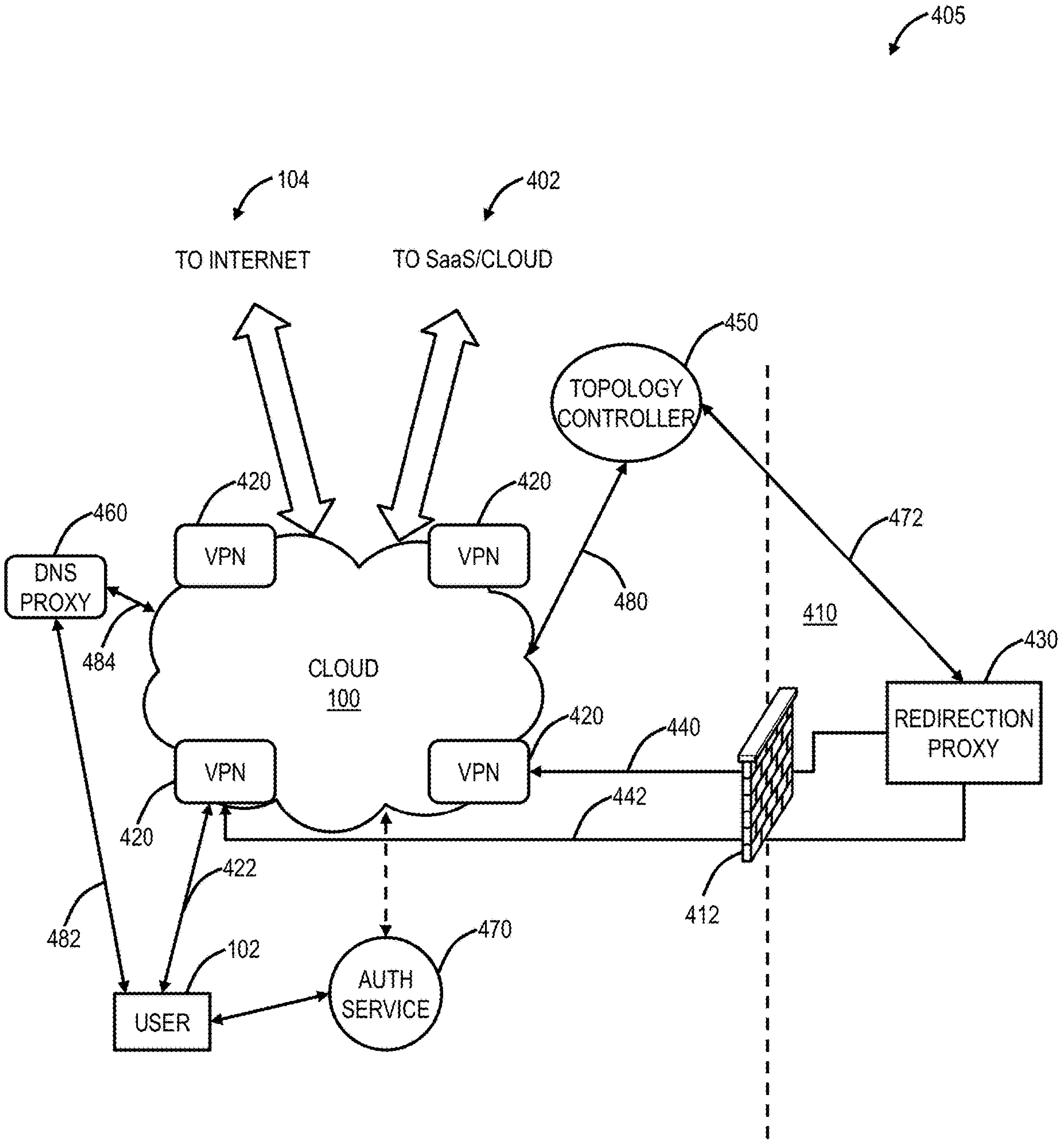

405
104
402
TO INTERNET
TO SaaS/CLOUD
450
TOPOLOGY CONTROLLER
420
420
460
DNS PROXY
VPN
VPN
472
480
410
484
CLOUD
100
430
420
440
REDIRECTION PROXY
VPN
VPN
442
420
422
412
482
102
470
AUTH SERVICE
USER

FIG. 7

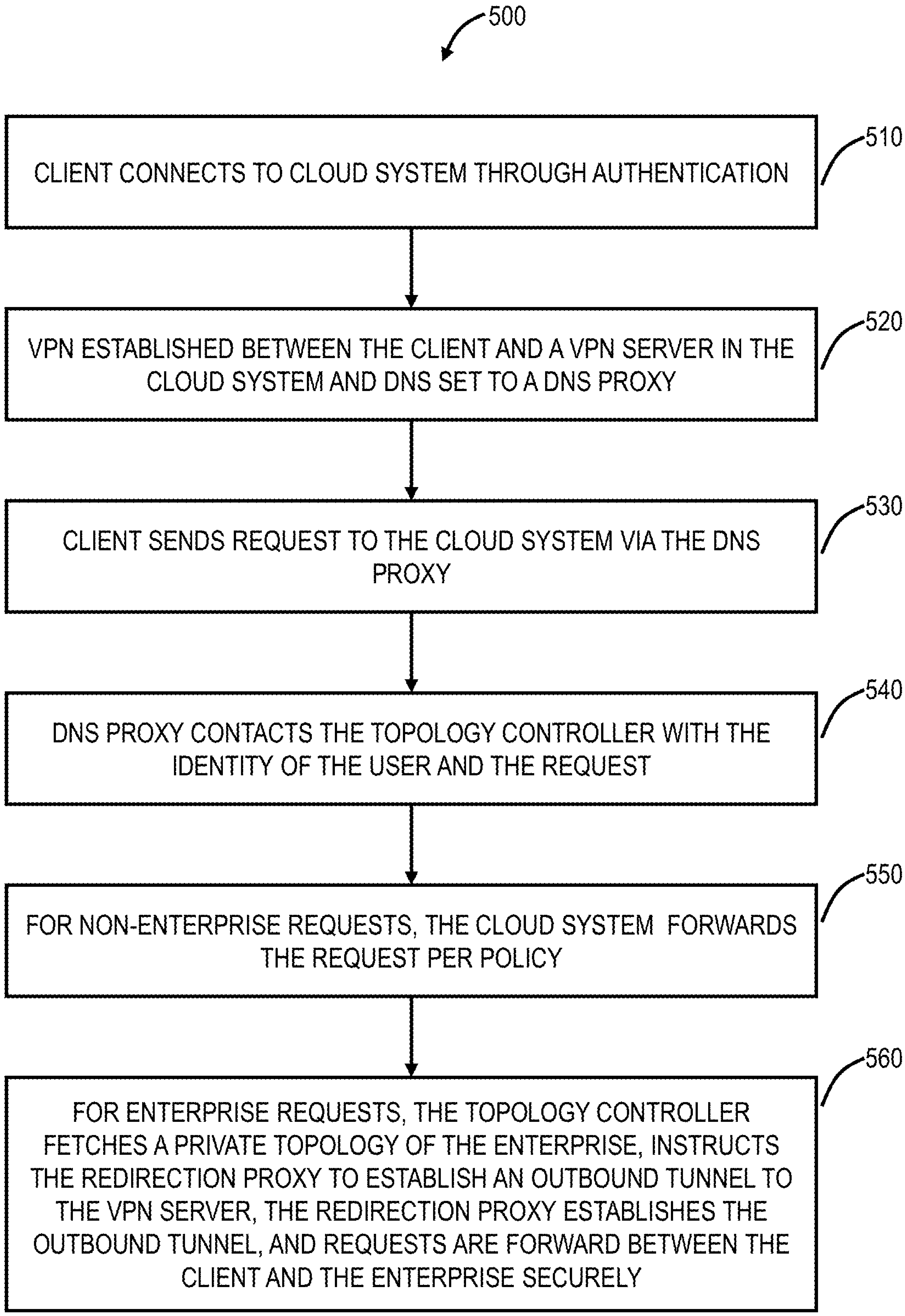
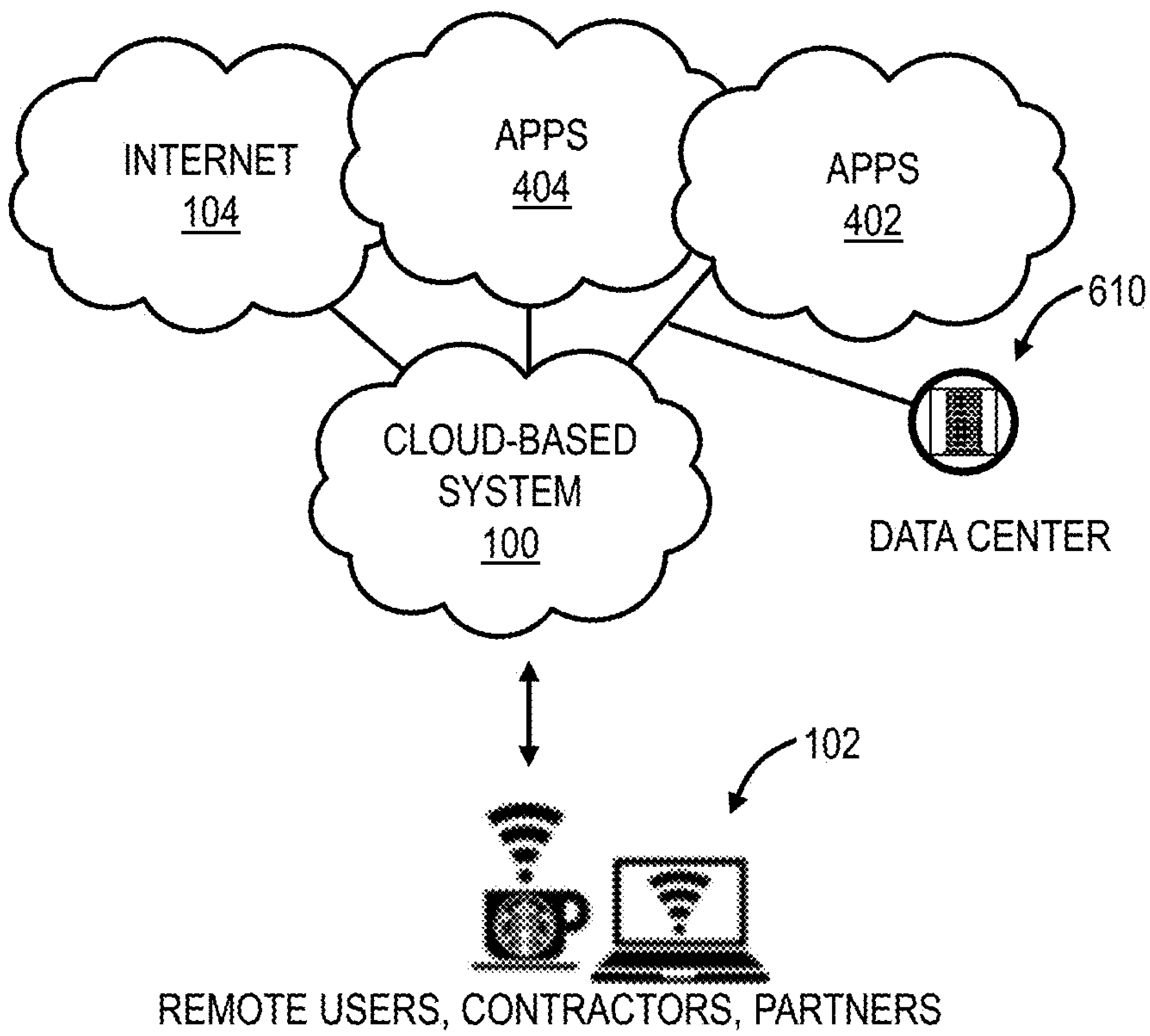

500
CLIENT CONNECTS TO CLOUD SYSTEM THROUGH AUTHENTICATION
510
VPN ESTABLISHED BETWEEN THE CLIENT AND A VPN SERVER IN THE CLOUD SYSTEM AND DNS SET TO A DNS PROXY
520
CLIENT SENDS REQUEST TO THE CLOUD SYSTEM VIA THE DNS PROXY
530
DNS PROXY CONTACTS THE TOPOLOGY CONTROLLER WITH THE IDENTITY OF THE USER AND THE REQUEST
540
FOR NON-ENTERPRISE REQUESTS, THE CLOUD SYSTEM FORWARDS THE REQUEST PER POLICY
550
FOR ENTERPRISE REQUESTS, THE TOPOLOGY CONTROLLER FETCHES A PRIVATE TOPOLOGY OF THE ENTERPRISE, INSTRUCTS THE REDIRECTION PROXY TO ESTABLISH AN OUTBOUND TUNNEL TO THE VPN SERVER, THE REDIRECTION PROXY ESTABLISHES THE OUTBOUND TUNNEL, AND REQUESTS ARE FORWARD BETWEEN THE CLIENT AND THE ENTERPRISE SECURELY
560

FIG. 8

FIG. 9


INTERNET
104
APPS
404
APPS
402
610
CLOUD-BASED
SYSTEM
100
DATA CENTER
102
REMOTE USERS, CONTRACTORS, PARTNERS

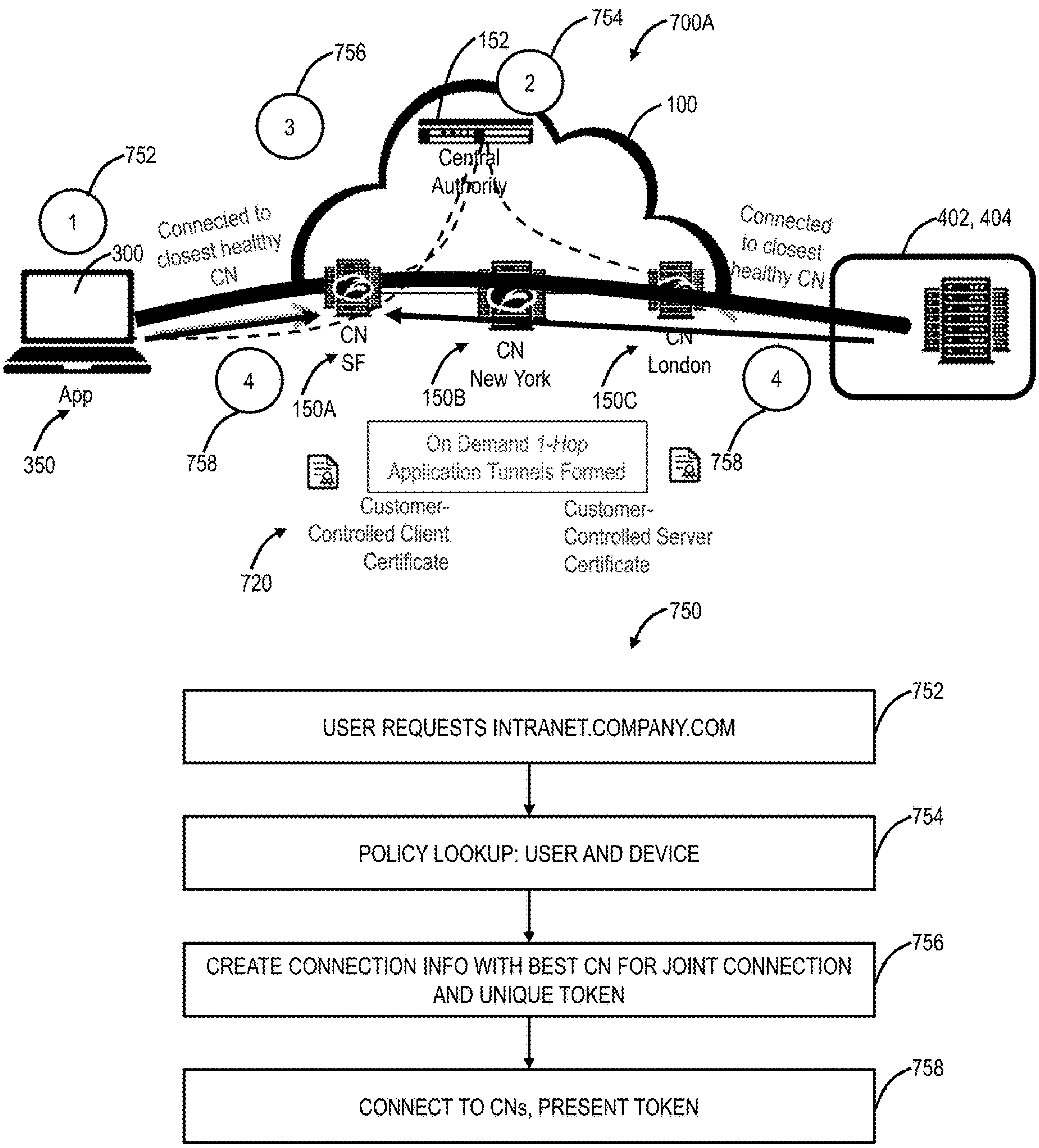

700A
754
152
2
756
3
100
Central Authority
752
1
300
Connected to closest healthy CN
Connected to closest healthy CN
402, 404
CN SF
CN New York
CN London
App
4
4
150A
150B
150C
350
758
758
On Demand 1-Hop Application Tunnels Formed
Customer-Controlled Client Certificate
Customer-Controlled Server Certificate
720
750
USER REQUESTS INTRANET.COMPANY.COM
752
POLICY LOOKUP: USER AND DEVICE
754
CREATE CONNECTION INFO WITH BEST CN FOR JOINT CONNECTION AND UNIQUE TOKEN
756
CONNECT TO CNs, PRESENT TOKEN
758

FIG. 10

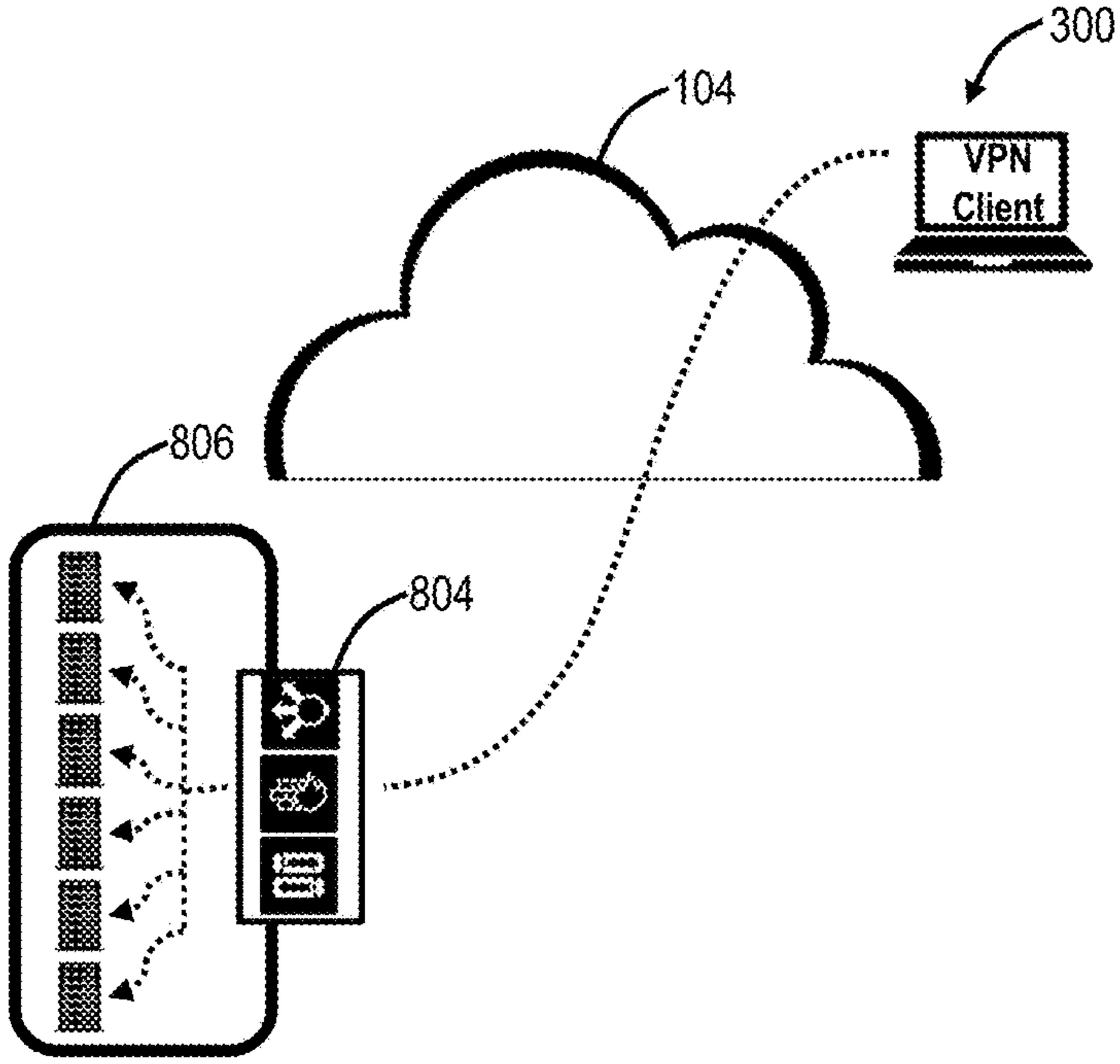


FIG. 11
(Prior Art)

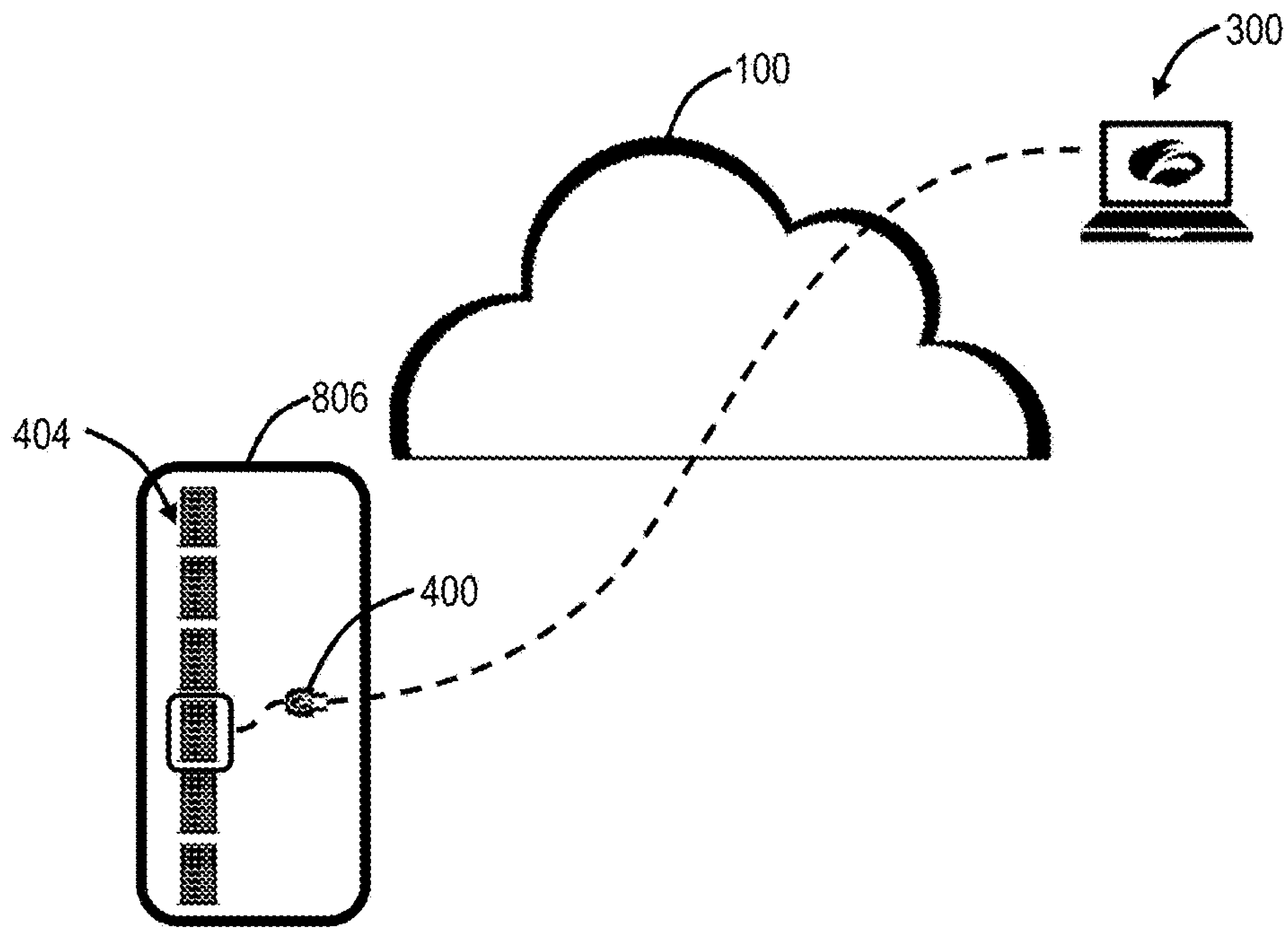


FIG. 12

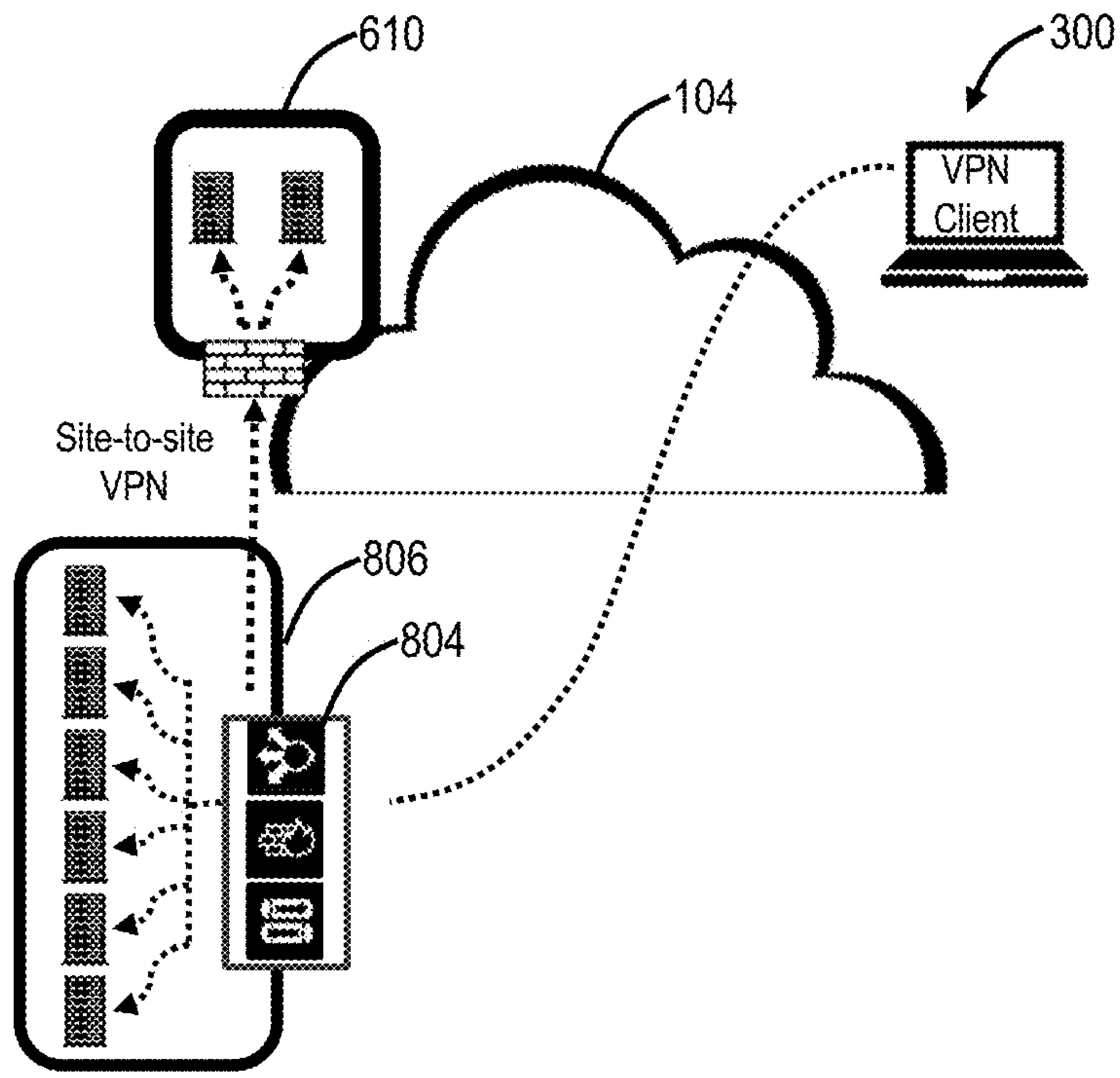


FIG. 13
(Prior Art)

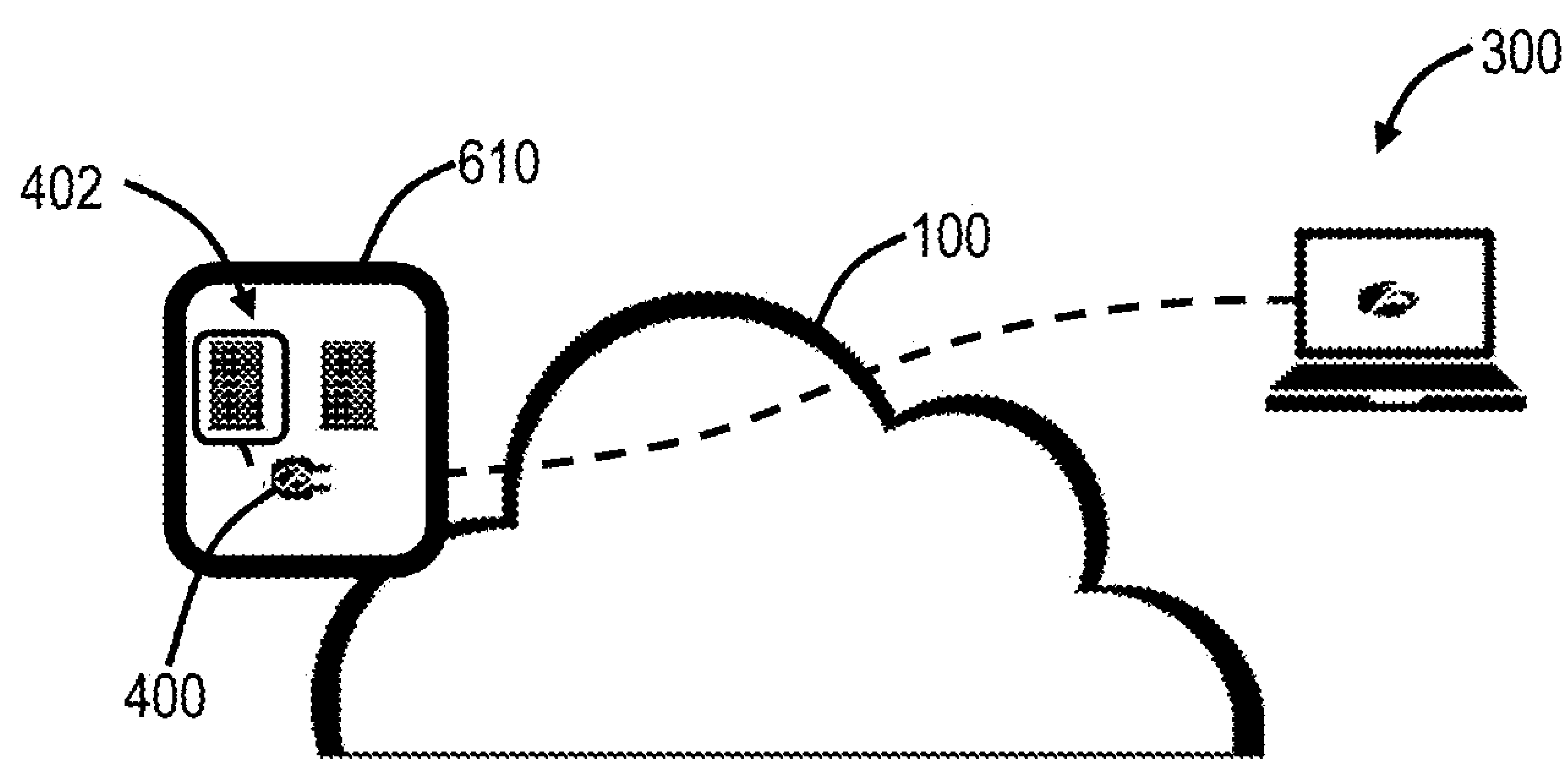


FIG. 14

FIG. 15
(Prior Art)

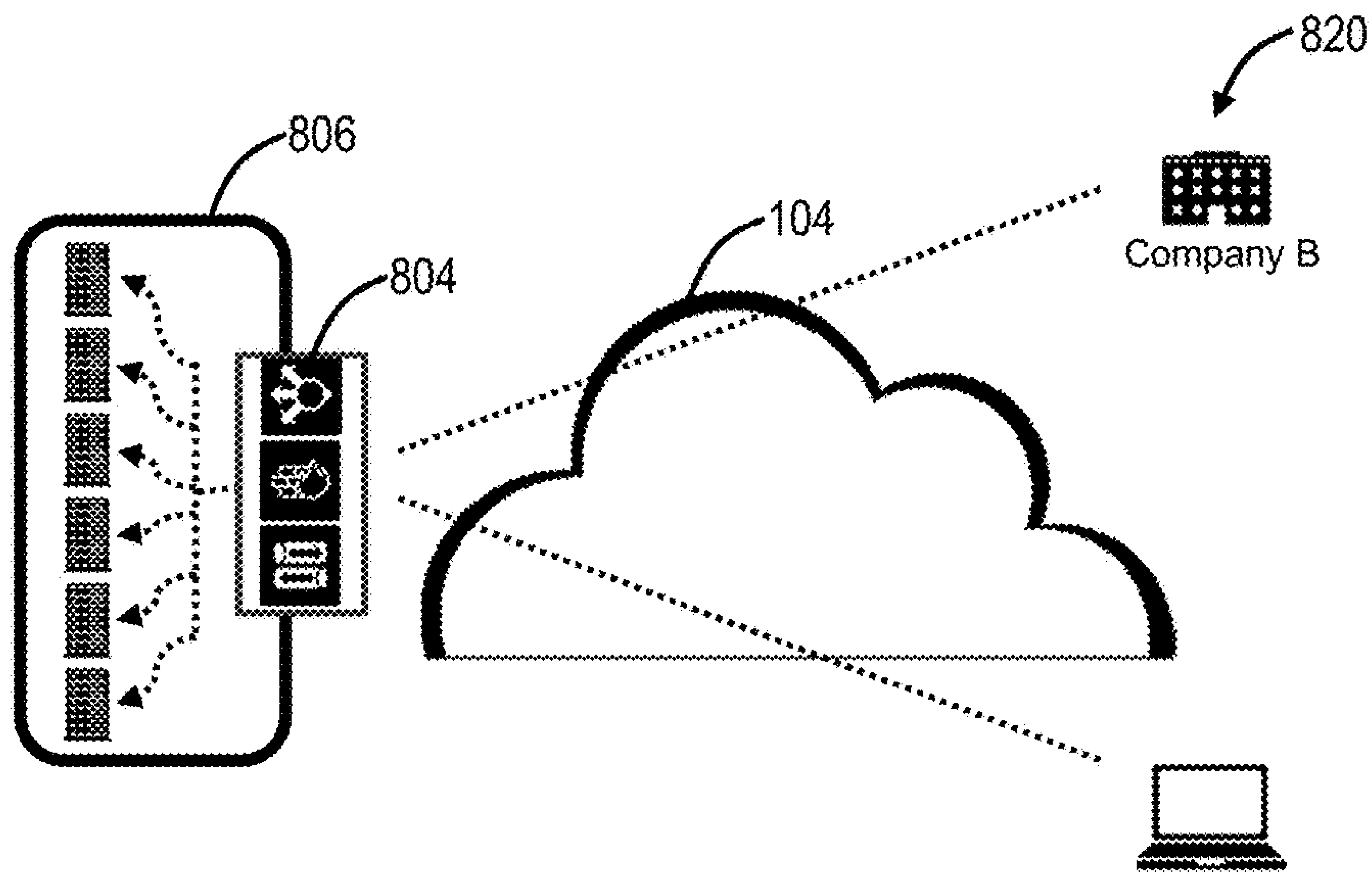


FIG. 16

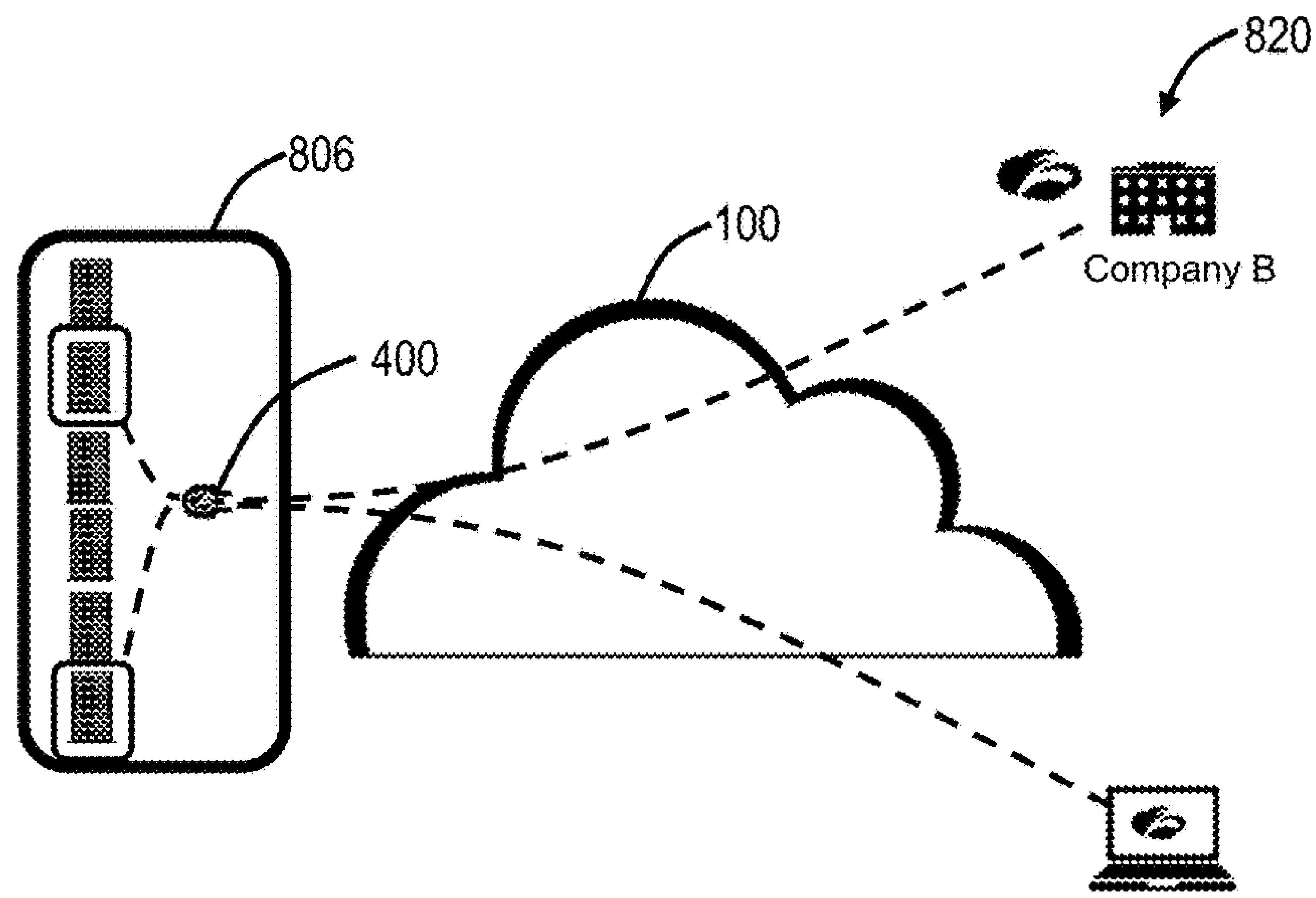


FIG. 17 (Prior Art)

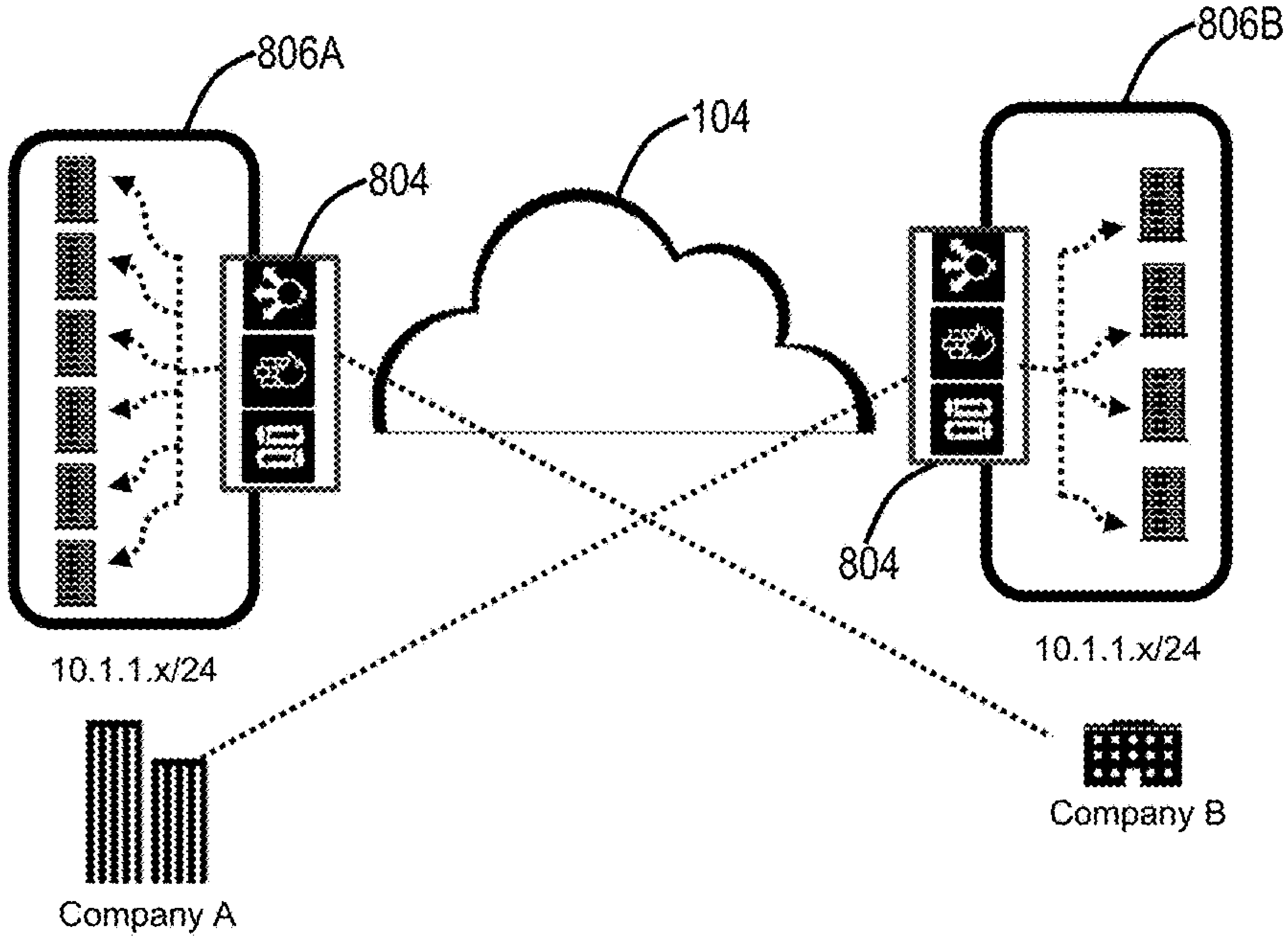
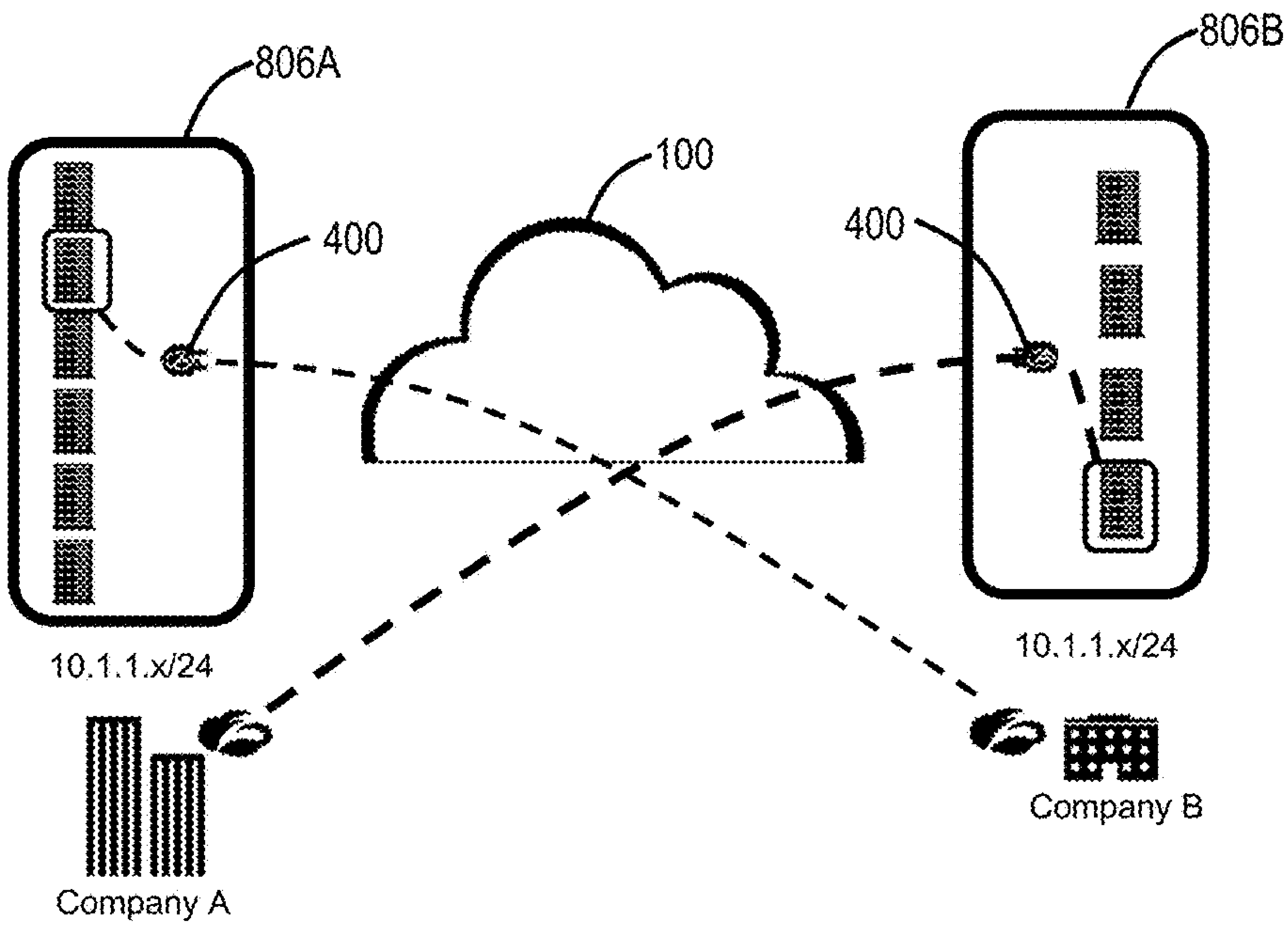


FIG. 18



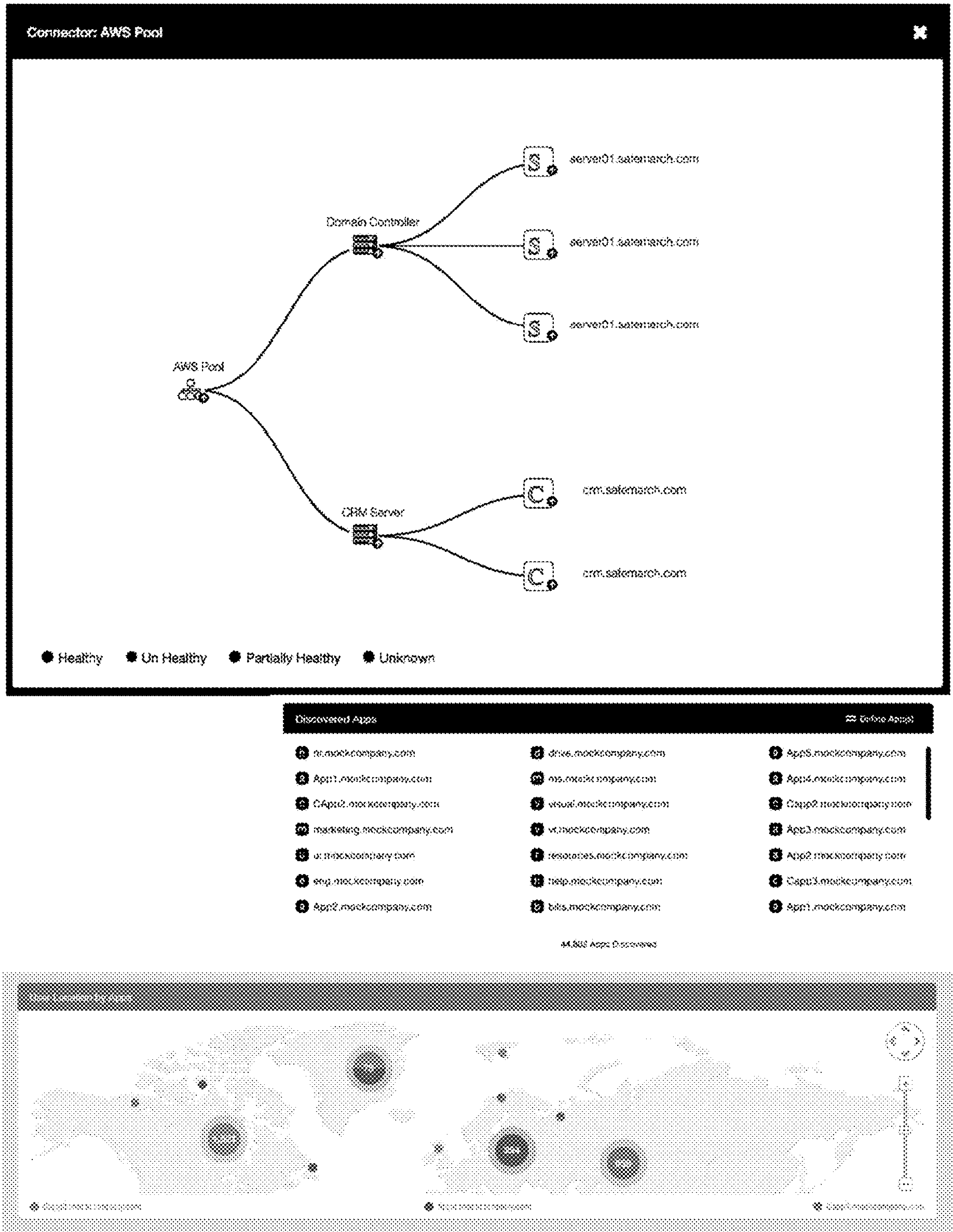

Connector: AWS Pool
server01.safemarch.com
Domain Controller
server01.safemarch.com
server01.safemarch.com
AWS Pool
crm.safemarch.com
CRM Server
crm.safemarch.com
Healthy
Un Healthy
Partially Healthy
Unknown
Discovered Apps
drive.mockcompany.com
marketing.mockcompany.com
resources.mockcompany.com
help.mockcompany.com
App3.mockcompany.com
App1.mockcompany.com

FIG. 19

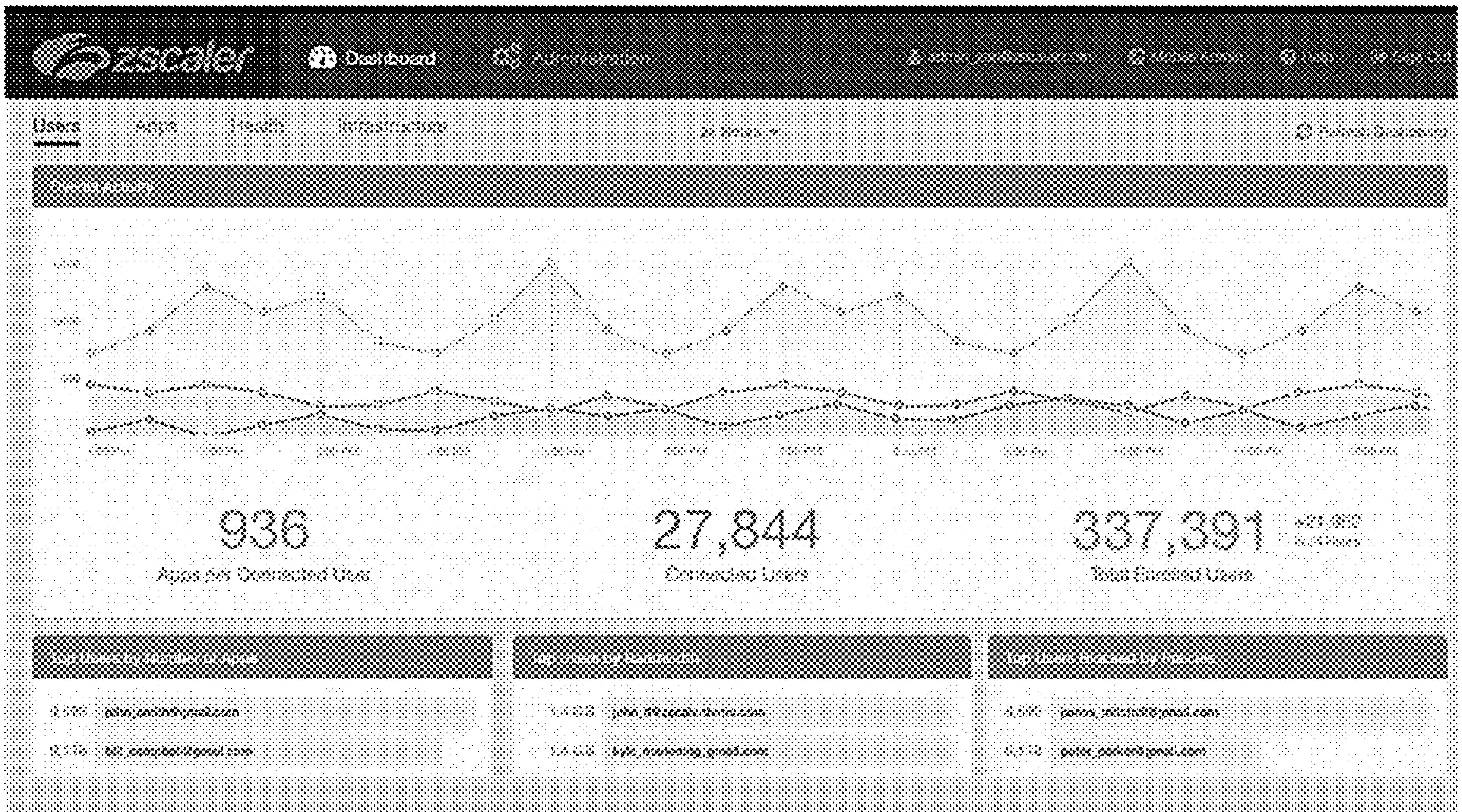

zscaler
Dashboard
Administration
Users
Apps
Health
Infrastructure
936
Apps per Connected User
27,844
Connected Users
337,391
Total Enrolled Users

FIG. 20

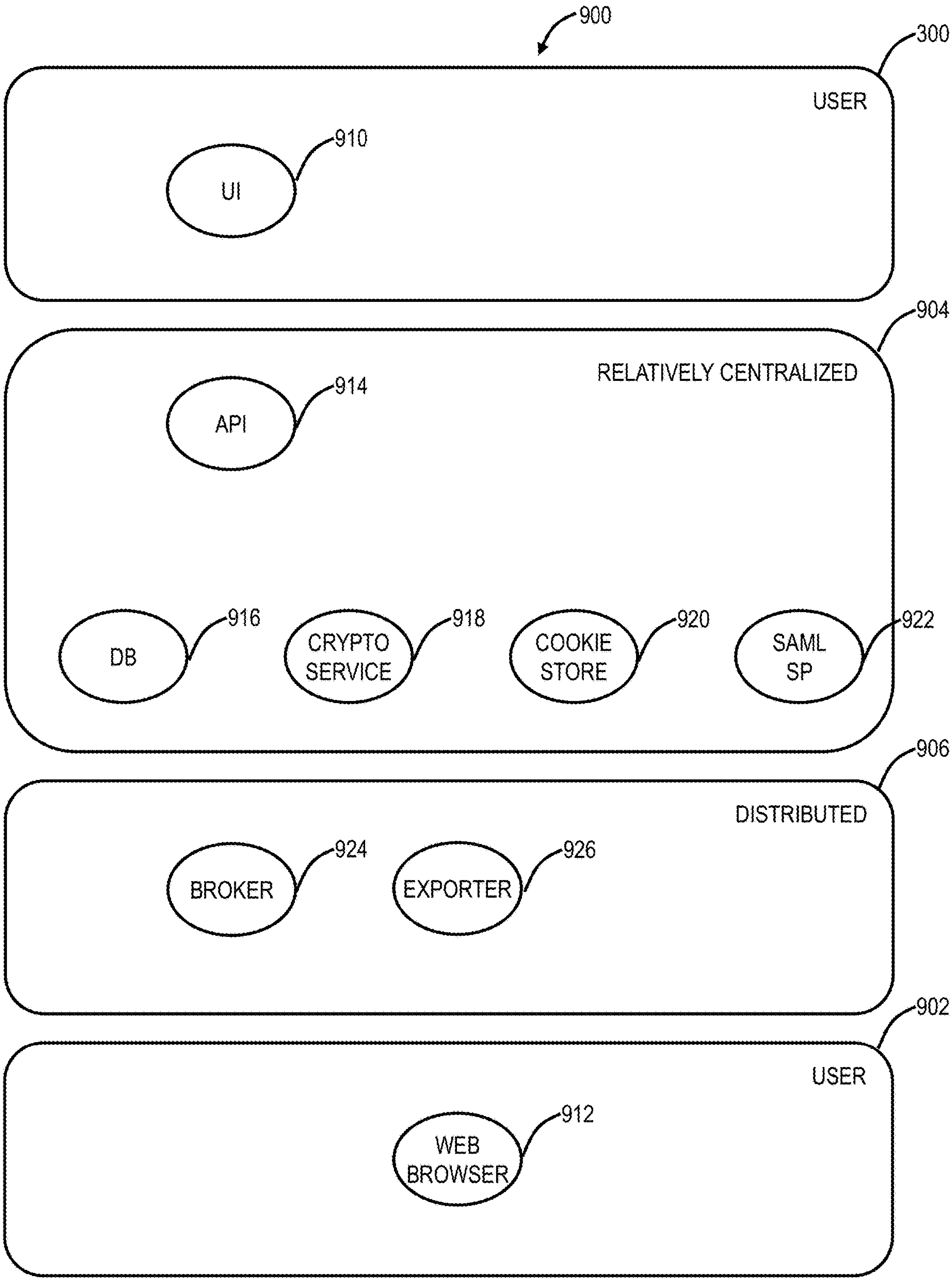

900
300
USER
UI
910
904
RELATIVELY CENTRALIZED
API
914
DB
916
CRYPTO SERVICE
918
COOKIE STORE
920
SAML SP
922
906
DISTRIBUTED
BROKER
924
EXPORTER
926
902
USER
WEB BROWSER
912

FIG. 21

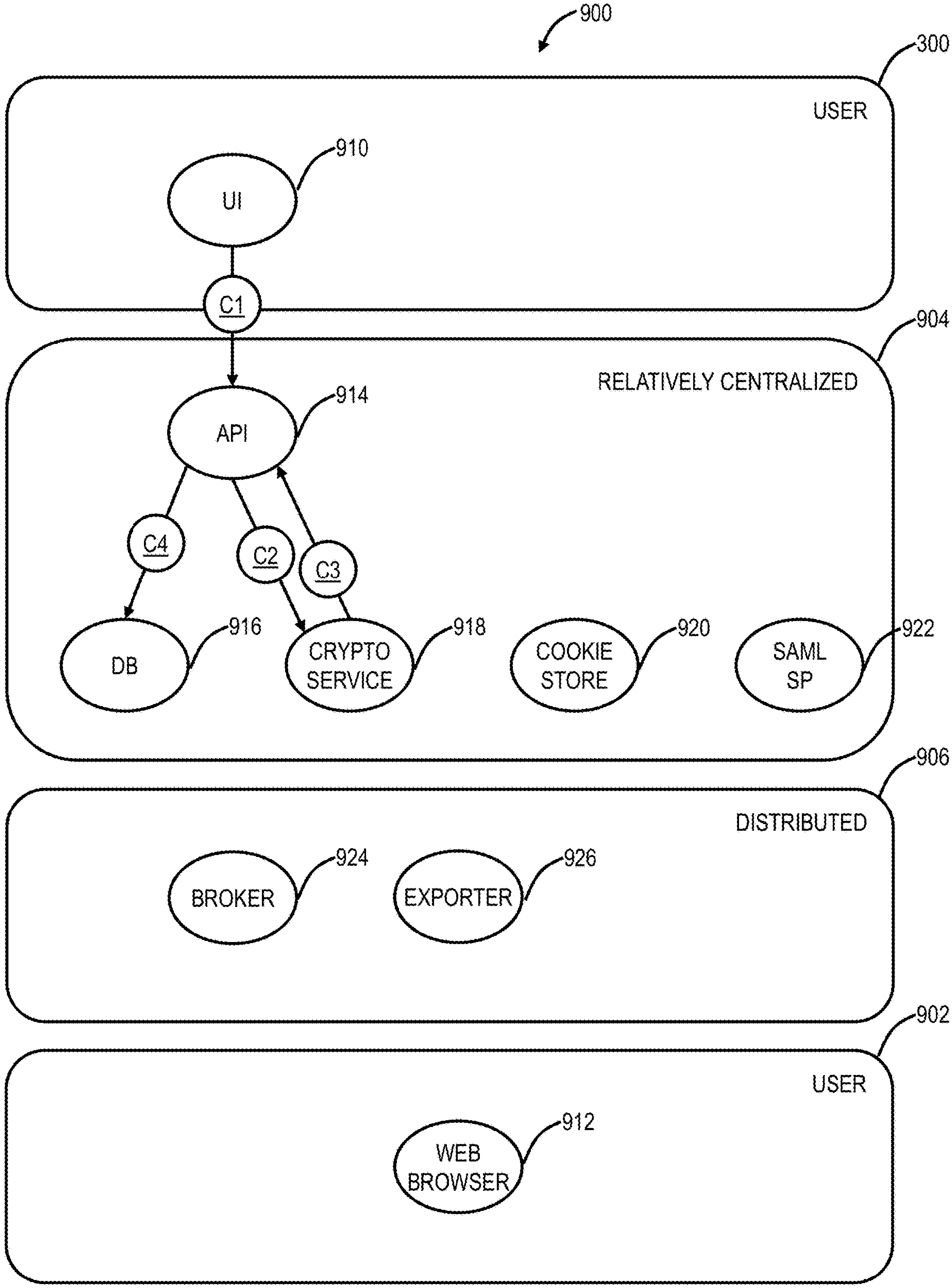

900
300
USER
UI
910
C1
904
RELATIVELY CENTRALIZED
API
914
C4
C2
C3
DB
916
CRYPTO SERVICE
918
COOKIE STORE
920
SAML SP
922
906
DISTRIBUTED
BROKER
924
EXPORTER
926
902
USER
WEB BROWSER
912

FIG. 22

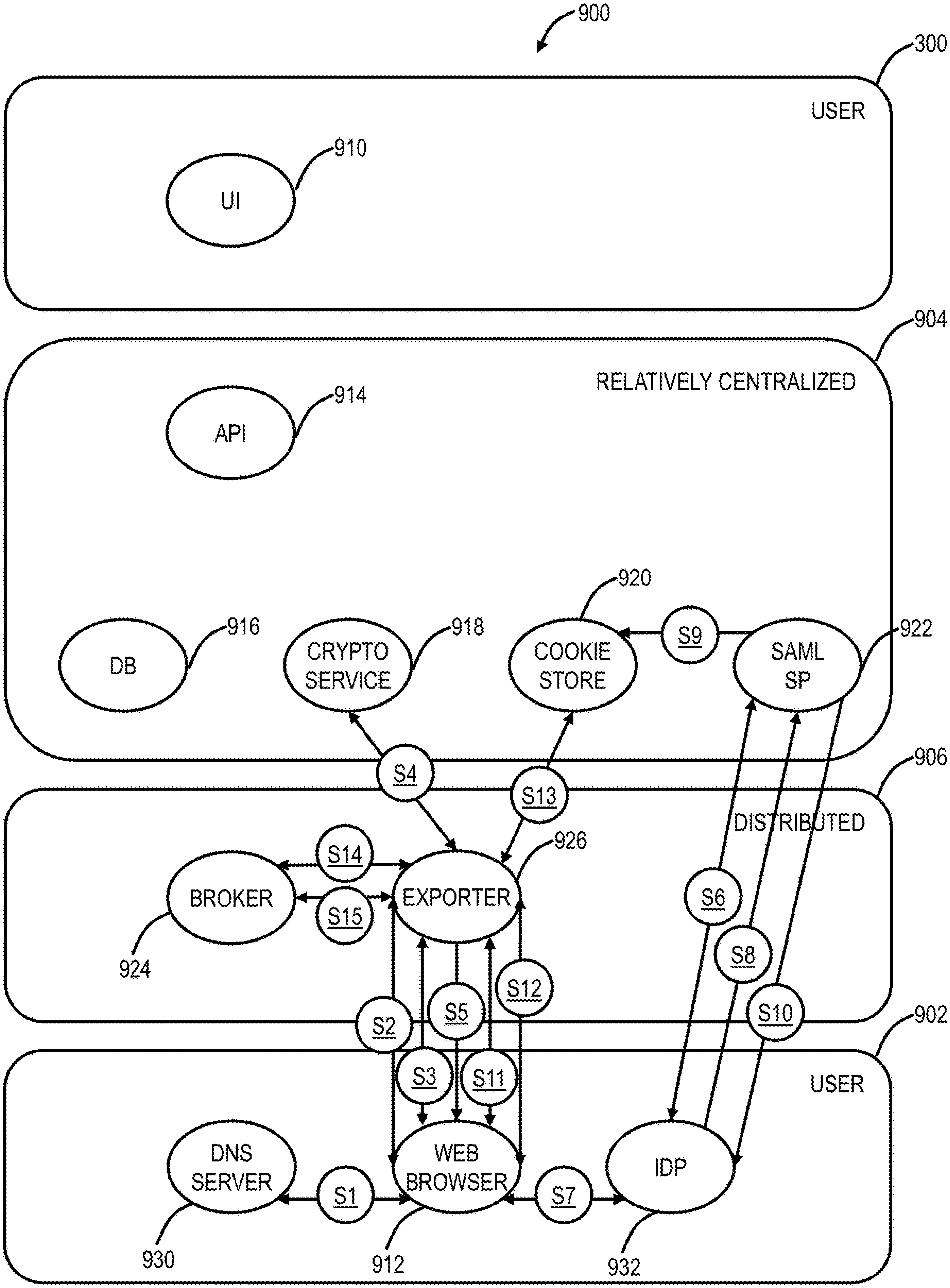

900
300
USER
910
UI
904
RELATIVELY CENTRALIZED
914
API
920
916
DB
918
CRYPTO SERVICE
COOKIE STORE
S9
SAML SP
922
S4
S13
906
DISTRIBUTED
926
S14
BROKER
S15
EXPORTER
S6
924
S8
S12
S5
S2
S10
902
S3
S11
USER
DNS SERVER
WEB BROWSER
IDP
S1
S7
930
912
932

FIG. 23

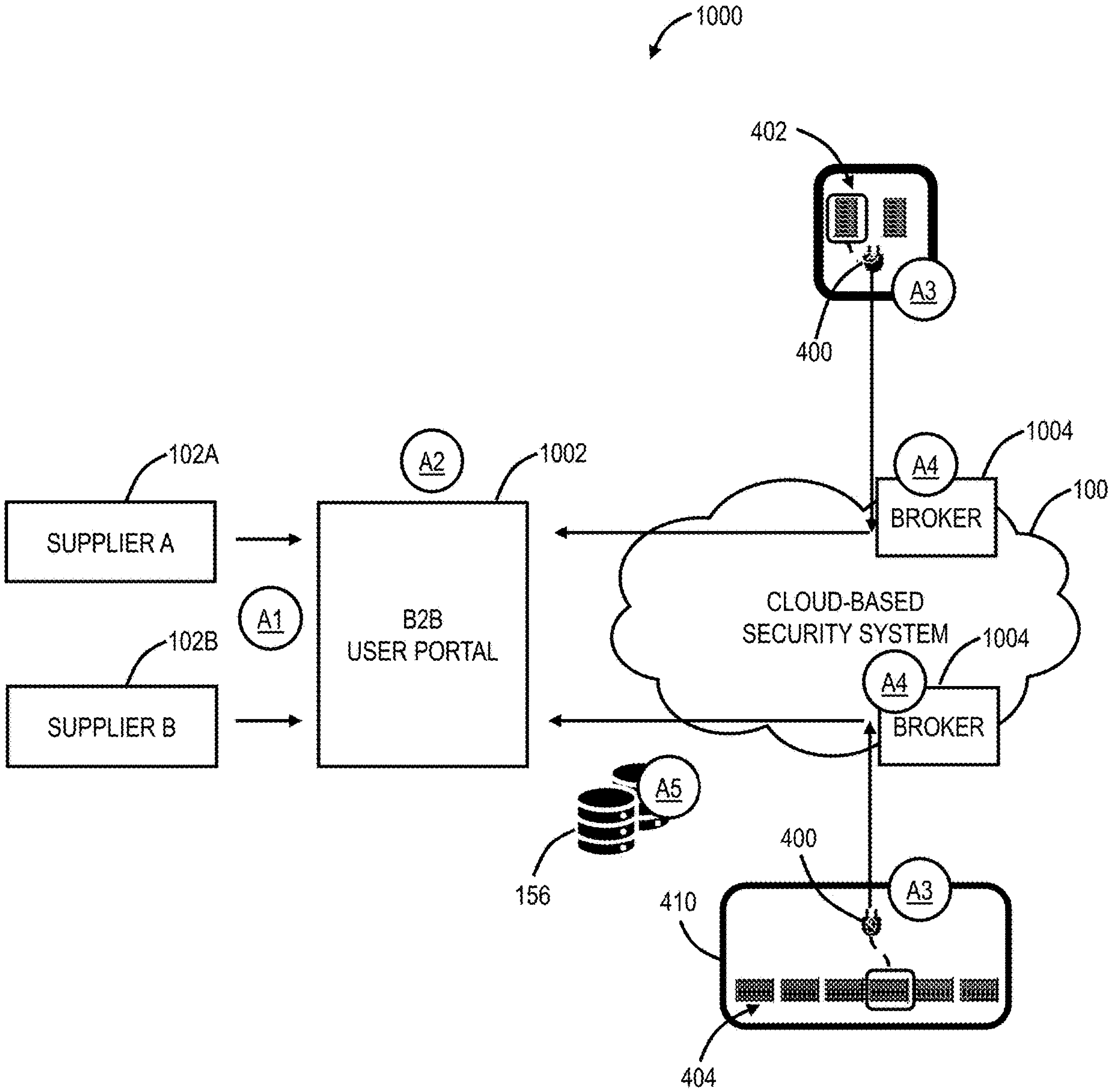

1000
402
A3
400
102A
A2
1002
A4
1004
100
SUPPLIER A
BROKER
A1
B2B
USER PORTAL
CLOUD-BASED
SECURITY SYSTEM
1004
102B
A4
SUPPLIER B
BROKER
A5
400
156
A3
410
404

FIG. 24

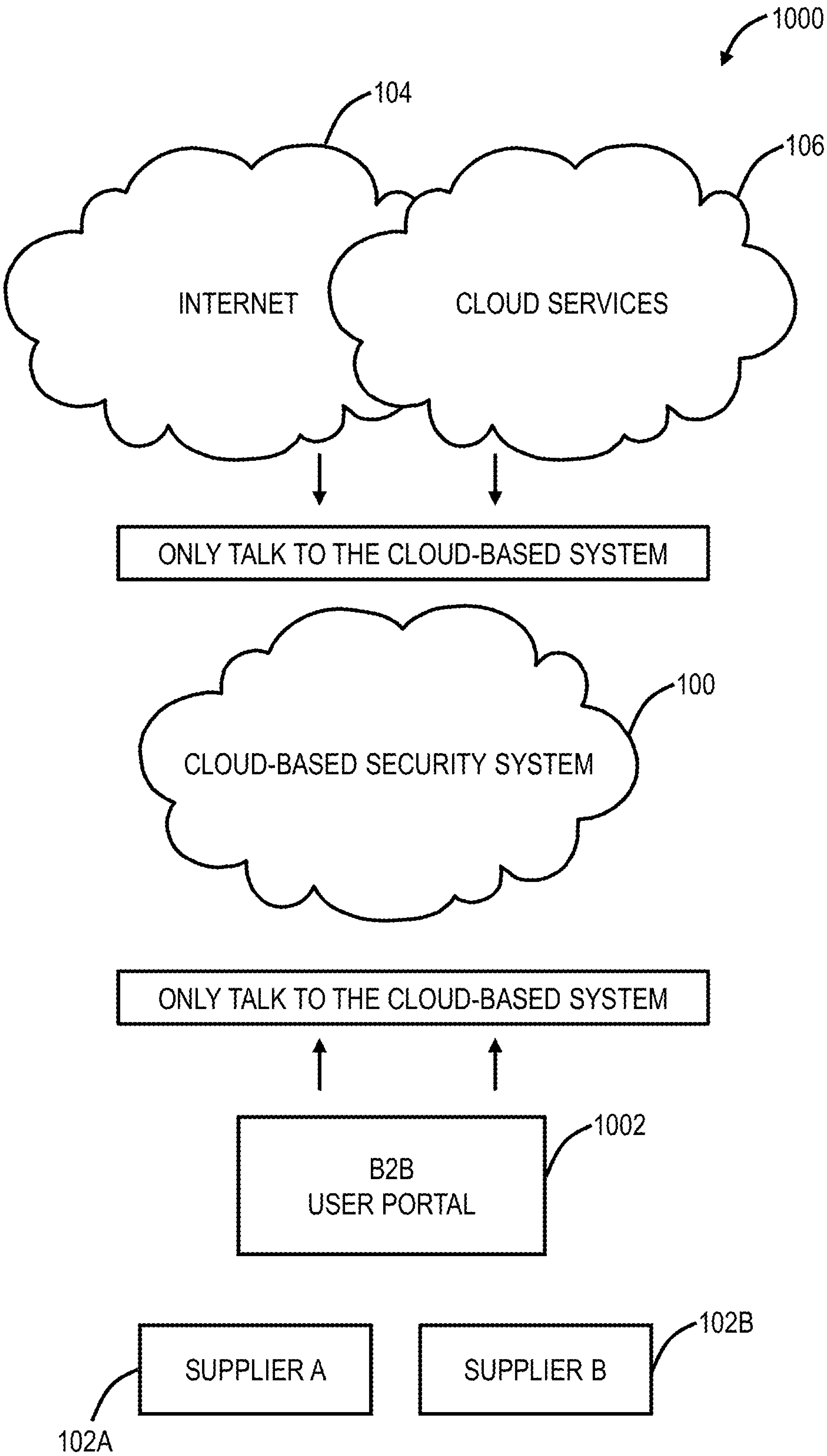

1000
104
106
INTERNET
CLOUD SERVICES
ONLY TALK TO THE CLOUD-BASED SYSTEM
100
CLOUD-BASED SECURITY SYSTEM
ONLY TALK TO THE CLOUD-BASED SYSTEM
1002
B2B
USER PORTAL
102B
SUPPLIER A
SUPPLIER B
102A

FIG. 25

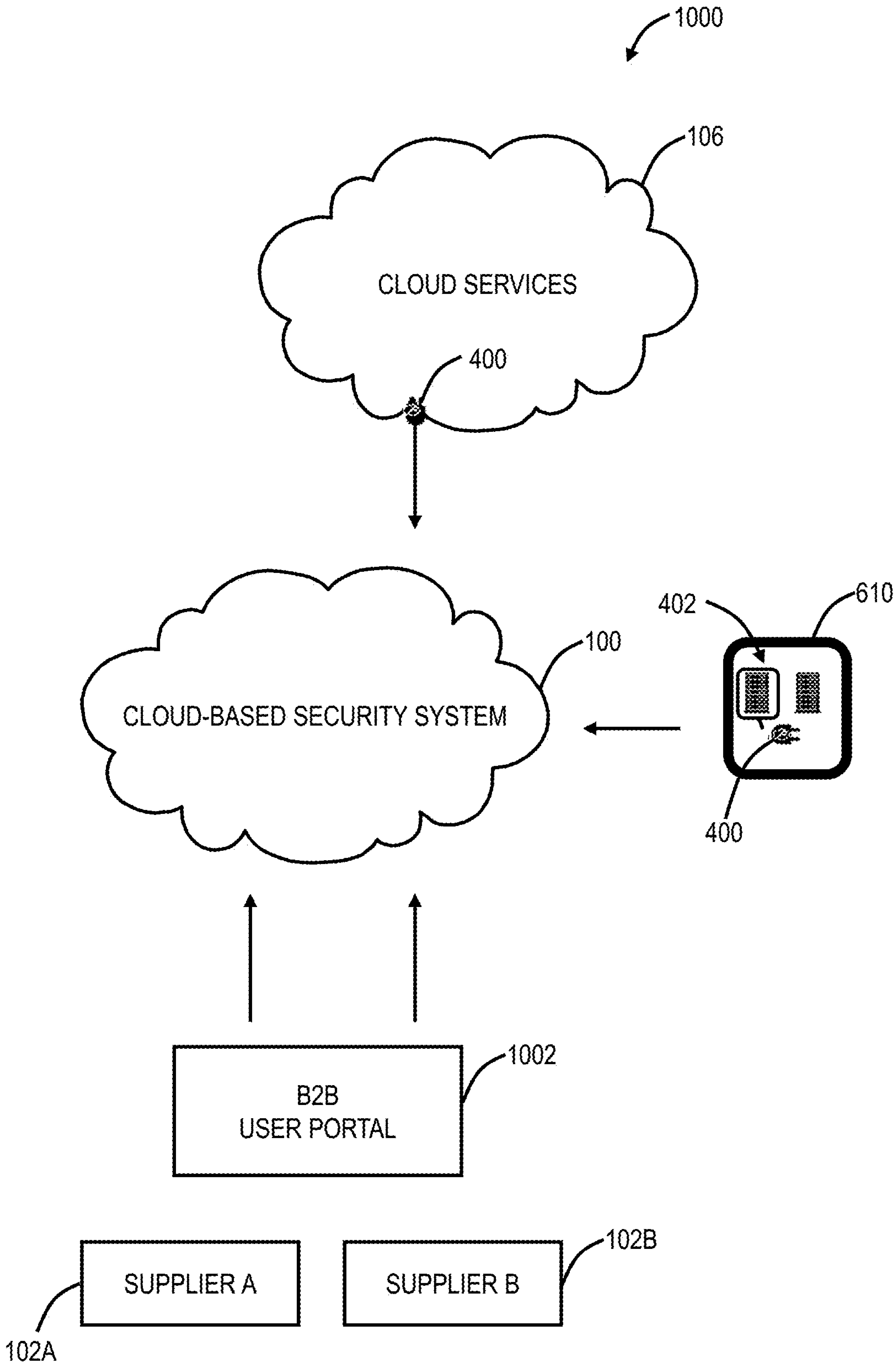

1000
106
CLOUD SERVICES
400
402
610
100
CLOUD-BASED SECURITY SYSTEM
400
1002
B2B
USER PORTAL
102B
SUPPLIER A
SUPPLIER B
102A

FIG. 26

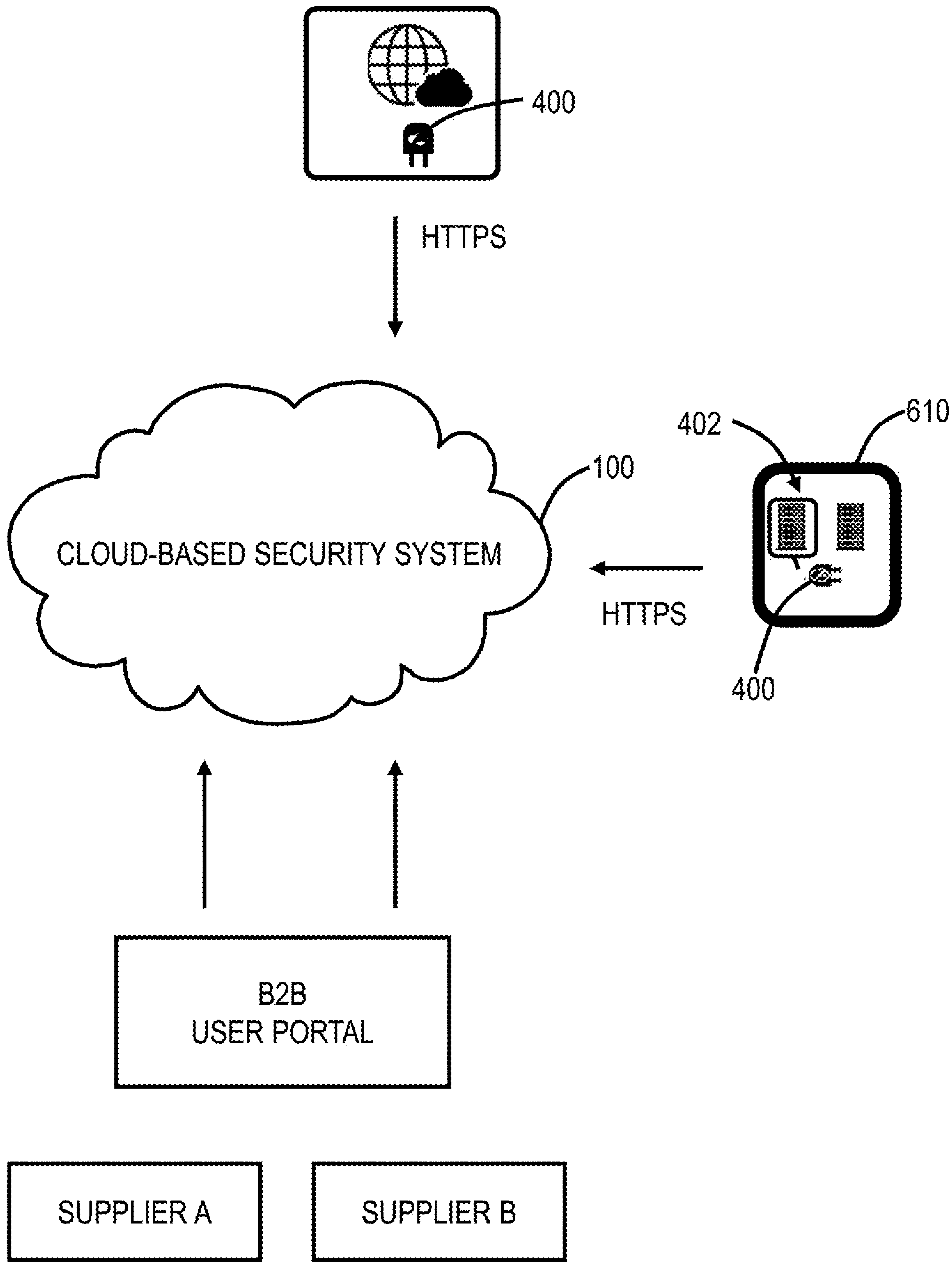

400
HTTPS
402
610
100
CLOUD-BASED SECURITY SYSTEM
HTTPS
400
B2B
USER PORTAL
SUPPLIER A
SUPPLIER B

FIG. 27

PROVIDING USERS SECURE ACCESS TO BUSINESS-TO-BUSINESS (B2B) APPLICATIONS

CROSS-REFERENCE TO RELATED APPLICATION(S)

The present patent/application is a continuation-in-part of U.S. patent application Ser. No. 16/800,307, filed Feb. 25, 2020, and entitled "Secure application access systems and methods," which is a continuation of U.S. patent application Ser. No. 15/986,874, filed May 23, 2018 (now U.S. Pat. No. 10,616,180, issued Apr. 7, 2020), and entitled "Clientless connection setup for cloud-based virtual private access systems and methods," which is a continuation-in-part of U.S. patent application Ser. No. 15/158,153 filed May 18, 2016 (now U.S. Pat. No. 10,375,024, issued Aug. 6, 2019), and entitled "Cloud-based virtual private access systems and methods," the contents of each are incorporated by reference herein in their entirety.

FIELD OF THE DISCLOSURE

The present disclosure generally relates to computer networking systems and methods. More particularly, the present disclosure relates to systems and methods for secure access for Business-to-Business (B2B) applications.

BACKGROUND OF THE DISCLOSURE

The cloud is redefining how businesses collaborate with customers. Business users (enterprises) need to provide secure and reliable application access to authorized business customers while maintaining superior user experience (UX). That is, there is no longer a consumer-app and a business-app; rather users, including businesses, are demanding a "customer app-like" experience when access B2B services. As described herein, the terms business apps, business services, B2B apps, B2B services, and B2B may be used interchangeably and include business-facing applications such as Supply Chain Management (SCM) applications, inventory management applications, ordering applications, financial applications, payroll applications, etc. That is, B2B apps are anything that an enterprise provides for its customers, partners, etc. B2B apps require exposure to the Internet, and ensuring this access to the B2B apps is secure and that the apps can scale is important. Most large enterprises have multiple business units, all with their own Information Technology (IT) infrastructure, requiring versatility.

As enterprises must expose these B2B services over the Internet and with the adoption of the public cloud, B2B apps now exist in multiple locations expanding the network points of presence. Traditional network security approaches like Secure Sockets Layer (SSL) Virtual Private Networks (VPNs), and Demilitarized Zones (DMZs) expose business services to attack by broadcasting them to the Internet, causing an expanded attack surface. While this is intended to make it easy for users to find apps, it means that the apps are exposed to potential bad actors, too. Teams should also realize that they do not own Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform's (GCP) network, so the idea of network security becomes irrelevant. Enterprises must adopt new technologies that help reduce their attack surface by negating the need for network connectivity.

Multiple business units create an administrative headache. Modern identity management solutions play a key role in verifying which business customers should have access to B2B apps. Disadvantageously, enterprises are often forced to either host identities themselves or deal with the inconsistent security practices of their business customers. Additionally, as applications are migrated from on-premises to the public cloud, teams should look to minimize any network and infrastructure changes. To provide the optimal customer experience, they should avoid the need to backhaul access through a central chokepoint and provide direct cloud connectivity instead.

Legacy technologies lack the scalability and availability that business requires. More and more revenue is coming from software sales. This increase demands that infrastructure can scale as fast as software can be built and ensure that revenue-driving B2B services are always available. The need for availability has led to the adoption of public clouds. The problem is that legacy, appliance-based infrastructure is still being used to access these software services, even though it was not built to scale elastically. Its limited availability creates chokepoints that impact access speeds. The boxes only handle a fixed amount of throughput and must be managed by the business unit. Application owners need to employ technologies that can be delivered as a cloud service.

Finally, poor business customer experience is bad for business. Business units must consider user experience as a competitive advantage and thus a priority. Online purchasing and mobile applications like Amazon, Uber, and Airbnb have created an expectation of a consumer app-like experience. Legacy technologies that require backhauling traffic to a data center (often located far away) create highly latent access and a poor experience for business customers. By bringing security to the edge via a cloud-delivered platform, business units can improve user experience.

BRIEF SUMMARY OF THE DISCLOSURE

The present disclosure relates to systems and methods for secure access for Business-to-Business (B2B) applications. Specifically, the present disclosure includes a cloud service that provides business customers with seamless, secure access to B2B applications. The service takes a Zero Trust Network Access (ZTNA) approach that uses business policy to reduce the attack surface of applications, preventing them from being exposed to the internet. Only authenticated (supporting modern Security Assertion Markup Language (SAML)-based Identify Providers (IDP)) and authorized users can see or access the B2B applications. The service is hosted through a cloud-based system, which removes the need to spend time managing network appliances. This reduction in operating expenses helps to accelerate cloud initiatives. The cloud service automatically connects business customers to apps via the fastest route by leveraging the cloud-based system's presence for higher availability, reliability, and scale. The objectives include connectivity based on business policies, ease of identity management, agentless access and improved customer UX, location agnostic, elimination of any attack surface, comprehensive visibility, and cost reduction.

BRIEF DESCRIPTION OF THE DRAWINGS

The present disclosure is illustrated and described herein with reference to the various drawings, in which like reference numbers are used to denote like system components/method steps, as appropriate, and in which:

FIG. 1 is a network diagram of a cloud-based system **100** offering security as a service;

FIG. 2 is a network diagram of an example implementation of the cloud-based system;

FIG. 3 is a network diagram of the cloud-based system illustrating an application on the user devices with users configured to operate through the cloud-based system;

FIG. 4 is a block diagram of a server, which may be used in the cloud-based system, in other systems, or standalone;

FIG. 5 is a block diagram of a user device, which may be used with the cloud-based system or the like;

FIG. 6 is a network diagram of a Zero Trust Network Access (ZTNA) application utilizing the cloud-based system;

FIG. 7 is a network diagram of a VPN architecture for an intelligent, cloud-based global VPN;

FIG. 8 is a flowchart of a VPN process for an intelligent, cloud-based global VPN;

FIG. 9 is a network diagram illustrating the cloud-based system with private applications and data centers connected thereto to provide virtual private access through the cloud-based system;

FIG. 10 is a network diagram of a virtual private access network and a flowchart of a virtual private access process implemented thereon;

FIGS. 11 and 12 are network diagrams of a VPN configuration (FIG. 11) compared to virtual private access (FIG. 12) illustrating the differences therein;

FIGS. 13 and 14 are network diagrams of conventional private application access in the public cloud (FIG. 13) compared to private application in the public cloud with virtual private access (FIG. 14);

FIGS. 15 and 16 are network diagrams of conventional contractor/partner access (FIG. 15) of applications in the enterprise network compared to contractor/partner access (FIG. 16) of the applications with virtual private access;

FIGS. 17 and 18 are network diagrams of a conventional network setup to share data between two companies (FIG. 17) such as for Merger and Acquisition (M&A) purposes or the like compared to a network setup using virtual private access (FIG. 18);

FIGS. 19 and 20 are screenshots of Graphical User Interfaces (GUIs) for administrator access to the virtual private access with FIG. 19 illustrating a GUI of network auto-discovery and FIG. 20 illustrating a GUI for reporting;

FIG. 21 is a block diagram of logical components of a clientless connection approach for virtual private access;

FIG. 22 is a block diagram of the logical components of the clientless connection approach illustrating the steps for configuration;

FIG. 23 is a block diagram of the logical components of the clientless connection approach illustrating the steps for a connection setup subsequent to the configuration;

FIG. 24 is a network diagram of a secure access for B2B applications network including the cloud-based system interconnecting users (e.g., suppliers) with the applications;

FIG. 25 is a network diagram of the secure access for B2B applications network illustrating a unique approach to ZTNA, removing exposure to the Internet;

FIG. 26 is a network diagram of the secure access for B2B applications network illustrating scalability and agility; and

FIG. 27 is a network diagram of the secure access for B2B applications network illustrating modernizing legacy applications.

DETAILED DESCRIPTION OF THE DISCLOSURE

Again, the present disclosure relates to systems and methods for secure access for Business-to-Business (B2B) applications. Specifically, the present disclosure includes a cloud service that provides business customers with seamless, secure access to B2B applications. The service takes a Zero Trust Network Access (ZTNA) approach that uses business policy to reduce the attack surface of applications, preventing them from being exposed to the internet. Only authenticated (supporting modern Security Assertion Markup Language (SAML)-based Identify Providers (IDP)) and authorized users can see or access the B2B applications. The service is hosted through a cloud-based system, which removes the need to spend time managing network appliances. This reduction in operating expenses helps to accelerate cloud initiatives. The cloud service automatically connects business customers to apps via the fastest route by leveraging the cloud-based system's presence for higher availability, reliability, and scale. The objectives include connectivity based on business policies, ease of identity management, agentless access and improved customer UX, location agnostic, elimination of any attack surface, comprehensive visibility, and cost reduction.

Also, systems and methods for cloud-based virtual private access of networked applications are described. At a high level, the systems and methods dynamically create a connection through a secure tunnel between three entities: an end-point, a cloud, and an on-premises redirection proxy. The connection between the cloud and on-premises proxy is dynamic, on-demand and orchestrated by the cloud. A key feature of the systems and methods is its security at the edge—there is no need to punch any holes in the existing on-premises firewall. The redirection proxy inside the enterprise (on premises) "dials out" and connects to the cloud as if too were an end-point. This on-demand dial-out capability and tunneling authenticated traffic back to the enterprise is a key differentiator of the systems and methods.

The paradigm of the virtual private access systems and methods is to give users network access to get to an application, not to the entire network. If a user is not authorized to get the application, the user should not be able to even see that it exists, much less access it. The virtual private access systems and methods provide a new approach to deliver secure access by decoupling applications from the network, instead providing access with a lightweight software connector, in front of the applications, an application on the user device, a central authority to push policy, and a cloud to stitch the applications and the software connectors together, on a per-user, per-application basis.

With the virtual private access, users can only see the specific applications allowed by policy. Everything else is "invisible" or "dark" to them. Because the virtual private access separates the application from the network, the physical location of the application becomes irrelevant—if applications are located in more than one place, the user is automatically directed to the instance that will give them the best performance. The virtual private access also dramatically reduces configuration complexity, such as policies/firewalls in the data centers. Enterprises can, for example, move applications to Amazon Web Services or Microsoft Azure, and take advantage of the elasticity of the cloud, making private, internal applications behave just like the marketing leading enterprise applications. Advantageously, there is no hardware to buy or deploy because the virtual private access is a service offering to users and enterprises.

Example Cloud-Based System Architecture

FIG. 1 is a network diagram of a cloud-based system **100** offering security as a service. Specifically, the cloud-based system **100** can offer a Secure Internet and Web Gateway as a service to various users **102**, as well as other cloud services. In this manner, the cloud-based system **100** is located between the users **102** and the Internet as well as any cloud services **106** (or applications) accessed by the users **102**. As such, the cloud-based system **100** provides inline monitoring inspecting traffic between the users **102**, the Internet **104**, and the cloud services **106**, including Secure Sockets Layer (SSL) traffic. The cloud-based system **100** can offer access control, threat prevention, data protection, etc. The access control can include a cloud-based firewall, cloud-based intrusion detection, Uniform Resource Locator (URL) filtering, bandwidth control, Domain Name System (DNS) filtering, etc. The threat prevention can include cloud-based intrusion prevention, protection against advanced threats (malware, spam, Cross-Site Scripting (XSS), phishing, etc.), cloud-based sandbox, antivirus, DNS security, etc. The data protection can include Data Loss Prevention (DLP), cloud application security such as via a Cloud Access Security Broker (CASB), file type control, etc.

The cloud-based firewall can provide Deep Packet Inspection (DPI) and access controls across various ports and protocols as well as being application and user aware. The URL filtering can block, allow, or limit website access based on policy for a user, group of users, or entire organization, including specific destinations or categories of URLs (e.g., gambling, social media, etc.). The bandwidth control can enforce bandwidth policies and prioritize critical applications such as relative to recreational traffic. DNS filtering can control and block DNS requests against known and malicious destinations.

The cloud-based intrusion prevention and advanced threat protection can deliver full threat protection against malicious content such as browser exploits, scripts, identified botnets and malware callbacks, etc. The cloud-based sandbox can block zero-day exploits (just identified) by analyzing unknown files for malicious behavior. Advantageously, the cloud-based system **100** is multi-tenant and can service a large volume of the users **102**. As such, newly discovered threats can be promulgated throughout the cloud-based system **100** for all tenants practically instantaneously. The antivirus protection can include antivirus, antispyware, antimalware, etc. protection for the users **102**, using signatures sourced and constantly updated. The DNS security can identify and route command-and-control connections to threat detection engines for full content inspection.

The DLP can use standard and/or custom dictionaries to continuously monitor the users **102**, including compressed and/or SSL-encrypted traffic. Again, being in a cloud implementation, the cloud-based system **100** can scale this monitoring with near-zero latency on the users **102**. The cloud application security can include CASB functionality to discover and control user access to known and unknown cloud services **106**. The file type controls enable true file type control by the user, location, destination, etc. to determine which files are allowed or not.

For illustration purposes, the users **102** of the cloud-based system **100** can include a mobile device **110**, a headquarters (HQ) **112** which can include or connect to a data center (DC) **114**, Internet of Things (IoT) devices **116**, a branch office/remote location **118**, etc., and each includes one or more user devices (an example user device **300** is illustrated in FIG. **5**). The devices **110**, **116**, and the locations **112**, **114**, **118** are shown for illustrative purposes, and those skilled in the art will recognize there are various access scenarios and other users **102** for the cloud-based system **100**, all of which are contemplated herein. The users **102** can be associated with a tenant, which may include an enterprise, a corporation, an organization, etc. That is, a tenant is a group of users who share a common access with specific privileges to the cloud-based system **100**, a cloud service, etc. In an embodiment, the headquarters **112** can include an enterprise's network with resources in the data center **114**. The mobile device **110** can be a so-called road warrior, i.e., users that are off-site, on-the-road, etc. Those skilled in the art will recognize a user **102** has to use a corresponding user device **300** for accessing the cloud-based system **100** and the like, and the description herein may use the user **102** and/or the user device **300** interchangeably.

Further, the cloud-based system **100** can be multi-tenant, with each tenant having its own users **102** and configuration, policy, rules, etc. One advantage of the multi-tenancy and a large volume of users is the zero-day/zero-hour protection in that a new vulnerability can be detected and then instantly remediated across the entire cloud-based system **100**. The same applies to policy, rule, configuration, etc. changes—they are instantly remediated across the entire cloud-based system **100**. As well, new features in the cloud-based system **100** can also be rolled up simultaneously across the user base, as opposed to selective and time-consuming upgrades on every device at the locations **112**, **114**, **118**, and the devices **110**, **116**.

Logically, the cloud-based system **100** can be viewed as an overlay network between users (at the locations **112**, **114**, **118**, and the devices **110**, **116**) and the Internet **104** and the cloud services **106**. Previously, the IT deployment model included enterprise resources and applications stored within the data center **114** (i.e., physical devices) behind a firewall (perimeter), accessible by employees, partners, contractors, etc. on-site or remote via Virtual Private Networks (VPNs), etc. The cloud-based system **100** is replacing the conventional deployment model. The cloud-based system **100** can be used to implement these services in the cloud without requiring the physical devices and management thereof by enterprise IT administrators. As an ever-present overlay network, the cloud-based system **100** can provide the same functions as the physical devices and/or appliances regardless of geography or location of the users **102**, as well as independent of platform, operating system, network access technique, network access provider, etc.

There are various techniques to forward traffic between the users **102** at the locations **112**, **114**, **118**, and via the devices **110**, **116**, and the cloud-based system **100**. Typically, the locations **112**, **114**, **118** can use tunneling where all traffic is forward through the cloud-based system **100**. For example, various tunneling protocols are contemplated, such as Generic Routing Encapsulation (GRE), Layer Two Tunneling Protocol (L2TP), Internet Protocol (IP) Security (IPsec), customized tunneling protocols, etc. The devices **110**, **116**, when not at one of the locations **112**, **114**, **118** can use a local application that forwards traffic, a proxy such as via a Proxy Auto-Config (PAC) file, and the like. An application of the local application is the application **350** described in detail herein as a connector application. A key aspect of the cloud-based system **100** is all traffic between the users **102** and the Internet **104** or the cloud services **106** is via the cloud-based system **100**. As such, the cloud-based system **100** has visibility to enable various functions, all of which are performed off the user device in the cloud.

The cloud-based system **100** can also include a management system **120** for tenant access to provide global policy and configuration as well as real-time analytics. This enables IT administrators to have a unified view of user activity, threat intelligence, application usage, etc. For example, IT administrators can drill-down to a per-user level to understand events and correlate threats, to identify compromised devices, to have application visibility, and the like. The cloud-based system **100** can further include connectivity to an Identity Provider (IDP) **122** for authentication of the users **102** and to a Security Information and Event Management (SIEM) system **124** for event logging. The system **124** can provide alert and activity logs on a per-user **102** basis.

FIG. **2** is a network diagram of an example implementation of the cloud-based system **100**. In an embodiment, the cloud-based system **100** includes a plurality of enforcement nodes (EN) **150**, labeled as enforcement nodes **150-1**, **150-2**, **150**-N, interconnected to one another and interconnected to a central authority (CA) **152**. The nodes **150** and the central authority **152**, while described as nodes, can include one or more servers, including physical servers, virtual machines (VM) executed on physical hardware, etc. An example of a server is illustrated in FIG. **4**. The cloud-based system **100** further includes a log router **154** that connects to a storage cluster **156** for supporting log maintenance from the enforcement nodes **150**. The central authority **152** provide centralized policy, real-time threat updates, etc. and coordinates the distribution of this data between the enforcement nodes **150**. The enforcement nodes **150** provide an onramp to the users **102** and are configured to execute policy, based on the central authority **152**, for each user **102**. The enforcement nodes **150** can be geographically distributed, and the policy for each user **102** follows that user **102** as he or she connects to the nearest (or other criteria) enforcement node **150**. Of note, the cloud-based system is an external system meaning it is separate from tenant's private networks (enterprise networks) as well as from networks associated with the devices **110**, **116**, and locations **112**, **118**.

The enforcement nodes **150** are full-featured secure internet gateways that provide integrated internet security. They inspect all web traffic bi-directionally for malware and enforce security, compliance, and firewall policies, as described herein, as well as various additional functionality. In an embodiment, each enforcement node **150** has two main modules for inspecting traffic and applying policies: a web module and a firewall module. The enforcement nodes **150** are deployed around the world and can handle hundreds of thousands of concurrent users with millions of concurrent sessions. Because of this, regardless of where the users **102** are, they can access the Internet **104** from any device, and the enforcement nodes **150** protect the traffic and apply corporate policies. The enforcement nodes **150** can implement various inspection engines therein, and optionally, send sandboxing to another system. The enforcement nodes **150** include significant fault tolerance capabilities, such as deployment in active-active mode to ensure availability and redundancy as well as continuous monitoring.

In an embodiment, customer traffic is not passed to any other component within the cloud-based system **100**, and the enforcement nodes **150** can be configured never to store any data to disk. Packet data is held in memory for inspection and then, based on policy, is either forwarded or dropped. Log data generated for every transaction is compressed, tokenized, and exported over secure Transport Layer Security (TLS) connections to the log routers **154** that direct the logs to the storage cluster **156**, hosted in the appropriate geographical region, for each organization. In an embodiment, all data destined for or received from the Internet is processed through one of the enforcement nodes **150**. In another embodiment, specific data specified by each tenant, e.g., only email, only executable files, etc., is processed through one of the enforcement nodes **150**.

Each of the enforcement nodes **150** may generate a decision vector D=[d1, d2, . . . , dn] for a content item of one or more parts C=[c1, c2, . . . , cm]. Each decision vector may identify a threat classification, e.g., clean, spyware, malware, undesirable content, innocuous, spam email, unknown, etc. For example, the output of each element of the decision vector D may be based on the output of one or more data inspection engines. In an embodiment, the threat classification may be reduced to a subset of categories, e.g., violating, non-violating, neutral, unknown. Based on the subset classification, the enforcement node **150** may allow the distribution of the content item, preclude distribution of the content item, allow distribution of the content item after a cleaning process, or perform threat detection on the content item. In an embodiment, the actions taken by one of the enforcement nodes **150** may be determinative on the threat classification of the content item and on a security policy of the tenant to which the content item is being sent from or from which the content item is being requested by. A content item is violating if, for any part C=[c1, c2, . . . , cm] of the content item, at any of the enforcement nodes **150**, any one of the data inspection engines generates an output that results in a classification of "violating."

The central authority **152** hosts all customer (tenant) policy and configuration settings. It monitors the cloud and provides a central location for software and database updates and threat intelligence. Given the multi-tenant architecture, the central authority **152** is redundant and backed up in multiple different data centers. The enforcement nodes **150** establish persistent connections to the central authority **152** to download all policy configurations. When a new user connects to an enforcement node **150**, a policy request is sent to the central authority **152** through this connection. The central authority **152** then calculates the policies that apply to that user **102** and sends the policy to the enforcement node **150** as a highly compressed bitmap.

The policy can be tenant-specific and can include access privileges for users, websites and/or content that is disallowed, restricted domains, DLP dictionaries, etc. Once downloaded, a tenant's policy is cached until a policy change is made in the management system **120**. The policy can be tenant-specific and can include access privileges for users, websites and/or content that is disallowed, restricted domains, DLP dictionaries, etc. When this happens, all of the cached policies are purged, and the enforcement nodes **150** request the new policy when the user **102** next makes a request. In an embodiment, the enforcement node **150** exchange "heartbeats" periodically, so all enforcement nodes **150** are informed when there is a policy change. Any enforcement node **150** can then pull the change in policy when it sees a new request.

The cloud-based system **100** can be a private cloud, a public cloud, a combination of a private cloud and a public cloud (hybrid cloud), or the like. Cloud computing systems and methods abstract away physical servers, storage, networking, etc., and instead offer these as on-demand and elastic resources. The National Institute of Standards and Technology (NIST) provides a concise and specific definition which states cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing differs from the classic client-server model by providing applications from a server that are executed and managed by a client's web browser or the like, with no installed client version of an application required. Centralization gives cloud service providers complete control over the versions of the browser-based and other applications provided to clients, which removes the need for version upgrades or license management on individual client computing devices. The phrase "Software as a Service" (SaaS) is sometimes used to describe application programs offered through cloud computing. A common shorthand for a provided cloud computing service (or even an aggregation of all existing cloud services) is "the cloud." The cloud-based system **100** is illustrated herein as an example embodiment of a cloud-based system, and other implementations are also contemplated.

As described herein, the terms cloud services and cloud applications may be used interchangeably. The cloud service **106** is any service made available to users on-demand via the Internet, as opposed to being provided from a company's on-premises servers. A cloud application, or cloud app, is a software program where cloud-based and local components work together. The cloud-based system **100** can be utilized to provide example cloud services, including Zscaler Internet Access (ZIA), Zscaler Private Access (ZPA), and Zscaler Digital Experience (ZDX), all from Zscaler, Inc. (the assignee and applicant of the present application). Also, there can be multiple different cloud-based systems **100**, including ones with different architectures and multiple cloud services. The ZIA service can provide the access control, threat prevention, and data protection described above with reference to the cloud-based system **100**. ZPA can include access control, microservice segmentation, etc. The ZDX service can provide monitoring of user experience, e.g., Quality of Experience (QoE), Quality of Service (QoS), etc., in a manner that can gain insights based on continuous, inline monitoring. For example, the ZIA service can provide a user with Internet Access, and the ZPA service can provide a user with access to enterprise resources instead of traditional Virtual Private Networks (VPNs), namely ZPA provides Zero Trust Network Access (ZTNA). Those of ordinary skill in the art will recognize various other types of cloud services **106** are also contemplated. Also, other types of cloud architectures are also contemplated, with the cloud-based system **100** presented for illustration purposes.

User Device Application for Traffic Forwarding and Monitoring

FIG. **3** is a network diagram of the cloud-based system **100** illustrating an application **350** on user devices **300** with users **102** configured to operate through the cloud-based system **100**. Different types of user devices **300** are proliferating, including Bring Your Own Device (BYOD) as well as IT-managed devices. The conventional approach for a user device **300** to operate with the cloud-based system **100** as well as for accessing enterprise resources includes complex policies, VPNs, poor user experience, etc. The application **350** can automatically forward user traffic with the cloud-based system **100** as well as ensuring that security and access policies are enforced, regardless of device, location, operating system, or application. The application **350** automatically determines if a user **102** is looking to access the open Internet **104**, a SaaS app, or an internal app running in public, private, or the datacenter and routes mobile traffic through the cloud-based system **100**. The application **350** can support various cloud services, including ZIA, ZPA, ZDX, etc., allowing the best in class security with zero trust access to internal apps. As described herein, the application **350** can also be referred to as a connector application.

The application **350** is configured to auto-route traffic for seamless user experience. This can be protocol as well as application-specific, and the application **350** can route traffic with a nearest or best fit enforcement node **150**. Further, the application **350** can detect trusted networks, allowed applications, etc. and support secure network access. The application **350** can also support the enrollment of the user device **300** prior to accessing applications. The application **350** can uniquely detect the users **102** based on fingerprinting the user device **300**, using criteria like device model, platform, operating system, etc. The application **350** can support Mobile Device Management (MDM) functions, allowing IT personnel to deploy and manage the user devices **300** seamlessly. This can also include the automatic installation of client and SSL certificates during enrollment. Finally, the application **350** provides visibility into device and app usage of the user **102** of the user device **300**.

The application **350** supports a secure, lightweight tunnel between the user device **300** and the cloud-based system **100**. For example, the lightweight tunnel can be HTTP-based. With the application **350**, there is no requirement for PAC files, an IPSec VPN, authentication cookies, or user **102** setup.

Example Server Architecture

FIG. **4** is a block diagram of a server **200**, which may be used in the cloud-based system **100**, in other systems, or standalone. For example, the enforcement nodes **150** and the central authority **152** may be formed as one or more of the servers **200**. The server **200** may be a digital computer that, in terms of hardware architecture, generally includes a processor **202**, input/output (I/O) interfaces **204**, a network interface **206**, a data store **208**, and memory **210**. It should be appreciated by those of ordinary skill in the art that FIG. **4** depicts the server **200** in an oversimplified manner, and a practical embodiment may include additional components and suitably configured processing logic to support known or conventional operating features that are not described in detail herein. The components (**202**, **204**, **206**, **208**, and **210**) are communicatively coupled via a local interface **212**. The local interface **212** may be, for example, but not limited to, one or more buses or other wired or wireless connections, as is known in the art. The local interface **212** may have additional elements, which are omitted for simplicity, such as controllers, buffers (caches), drivers, repeaters, and receivers, among many others, to enable communications. Further, the local interface **212** may include address, control, and/or data connections to enable appropriate communications among the aforementioned components.

The processor **202** is a hardware device for executing software instructions. The processor **202** may be any custom made or commercially available processor, a Central Processing Unit (CPU), an auxiliary processor among several processors associated with the server **200**, a semiconductor-based microprocessor (in the form of a microchip or chipset), or generally any device for executing software instructions. When the server **200** is in operation, the processor **202** is configured to execute software stored within the memory **210**, to communicate data to and from the memory **210**, and to generally control operations of the server **200** pursuant to the software instructions. The I/O interfaces **204** may be used to receive user input from and/or for providing system output to one or more devices or components.

The network interface **206** may be used to enable the server **200** to communicate on a network, such as the Internet **104**. The network interface **206** may include, for example, an Ethernet card or adapter or a Wireless Local Area Network (WLAN) card or adapter. The network interface **206** may include address, control, and/or data connections to enable appropriate communications on the network. A data store **208** may be used to store data. The data store **208** may include any of volatile memory elements (e.g., random access memory (RAM, such as DRAM, SRAM, SDRAM, and the like)), nonvolatile memory elements (e.g., ROM, hard drive, tape, CDROM, and the like), and combinations thereof.

Moreover, the data store **208** may incorporate electronic, magnetic, optical, and/or other types of storage media. In one example, the data store **208** may be located internal to the server **200**, such as, for example, an internal hard drive connected to the local interface **212** in the server **200**. Additionally, in another embodiment, the data store **208** may be located external to the server **200** such as, for example, an external hard drive connected to the I/O interfaces **204** (e.g., SCSI or USB connection). In a further embodiment, the data store **208** may be connected to the server **200** through a network, such as, for example, a network-attached file server.

The memory **210** may include any of volatile memory elements (e.g., random access memory (RAM, such as DRAM, SRAM, SDRAM, etc.)), nonvolatile memory elements (e.g., ROM, hard drive, tape, CDROM, etc.), and combinations thereof. Moreover, the memory **210** may incorporate electronic, magnetic, optical, and/or other types of storage media. Note that the memory **210** may have a distributed architecture, where various components are situated remotely from one another but can be accessed by the processor **202**. The software in memory **210** may include one or more software programs, each of which includes an ordered listing of executable instructions for implementing logical functions. The software in the memory **210** includes a suitable Operating System (O/S) **214** and one or more programs **216**. The operating system **214** essentially controls the execution of other computer programs, such as the one or more programs **216**, and provides scheduling, input-output control, file and data management, memory management, and communication control and related services. The one or more programs **216** may be configured to implement the various processes, algorithms, methods, techniques, etc. described herein.

Example User Device Architecture

FIG. **5** is a block diagram of a user device **300**, which may be used with the cloud-based system **100** or the like. Specifically, the user device **300** can form a device used by one of the users **102**, and this may include common devices such as laptops, smartphones, tablets, netbooks, personal digital assistants, MP3 players, cell phones, e-book readers, IoT devices, servers, desktops, printers, televisions, streaming media devices, and the like. The user device **300** can be a digital device that, in terms of hardware architecture, generally includes a processor **302**, I/O interfaces **304**, a network interface **306**, a data store **308**, and memory **310**. It should be appreciated by those of ordinary skill in the art that FIG. **5** depicts the user device **300** in an oversimplified manner, and a practical embodiment may include additional components and suitably configured processing logic to support known or conventional operating features that are not described in detail herein. The components (**302**, **304**, **306**, **308**, and **302**) are communicatively coupled via a local interface **312**. The local interface **312** can be, for example, but not limited to, one or more buses or other wired or wireless connections, as is known in the art. The local interface **312** can have additional elements, which are omitted for simplicity, such as controllers, buffers (caches), drivers, repeaters, and receivers, among many others, to enable communications. Further, the local interface **312** may include address, control, and/or data connections to enable appropriate communications among the aforementioned components.

The processor **302** is a hardware device for executing software instructions. The processor **302** can be any custom made or commercially available processor, a CPU, an auxiliary processor among several processors associated with the user device **300**, a semiconductor-based microprocessor (in the form of a microchip or chipset), or generally any device for executing software instructions. When the user device **300** is in operation, the processor **302** is configured to execute software stored within the memory **310**, to communicate data to and from the memory **310**, and to generally control operations of the user device **300** pursuant to the software instructions. In an embodiment, the processor **302** may include a mobile optimized processor such as optimized for power consumption and mobile applications. The I/O interfaces **304** can be used to receive user input from and/or for providing system output. User input can be provided via, for example, a keypad, a touch screen, a scroll ball, a scroll bar, buttons, a barcode scanner, and the like. System output can be provided via a display device such as a Liquid Crystal Display (LCD), touch screen, and the like.

The network interface **306** enables wireless communication to an external access device or network. Any number of suitable wireless data communication protocols, techniques, or methodologies can be supported by the network interface **306**, including any protocols for wireless communication. The data store **308** may be used to store data. The data store **308** may include any of volatile memory elements (e.g., random access memory (RAM, such as DRAM, SRAM, SDRAM, and the like)), nonvolatile memory elements (e.g., ROM, hard drive, tape, CDROM, and the like), and combinations thereof. Moreover, the data store **308** may incorporate electronic, magnetic, optical, and/or other types of storage media.

The memory **310** may include any of volatile memory elements (e.g., random access memory (RAM, such as DRAM, SRAM, SDRAM, etc.)), nonvolatile memory elements (e.g., ROM, hard drive, etc.), and combinations thereof. Moreover, the memory **310** may incorporate electronic, magnetic, optical, and/or other types of storage media. Note that the memory **310** may have a distributed architecture, where various components are situated remotely from one another but can be accessed by the processor **302**. The software in memory **310** can include one or more software programs, each of which includes an ordered listing of executable instructions for implementing logical functions. In the example of FIG. **3**, the software in the memory **310** includes a suitable operating system **314** and programs **316**. The operating system **314** essentially controls the execution of other computer programs and provides scheduling, input-output control, file and data management, memory management, and communication control and related services. The programs **316** may include various applications, add-ons, etc. configured to provide end user functionality with the user device **300**. For example, example programs **316** may include, but not limited to, a web browser, social networking applications, streaming media applications, games, mapping and location applications, electronic mail applications, financial applications, and the like. In a typical example, the end-user typically uses one or more of the programs **316** along with a network such as the cloud-based system **100**.

Zero Trust Network Access Using the Cloud-Based System

FIG. **6** is a network diagram of a Zero Trust Network Access (ZTNA) application utilizing the cloud-based system **100**. For ZTNA, the cloud-based system **100** can dynamically create a connection through a secure tunnel between an endpoint (e.g., users **102**A, **102**B) that are remote and an on-premises connector **400** that is either located in cloud file shares and applications **402** and/or in an enterprise network **410** that includes enterprise file shares and applications **404**. The connection between the cloud-based system **100** and on-premises connector **400** is dynamic, on-demand, and orchestrated by the cloud-based system **100**. A key feature is its security at the edge—there is no need to punch any holes in the existing on-premises firewall. The connector **400** inside the enterprise (on-premises) "dials out" and connects to the cloud-based system **100** as if too were an endpoint. This on-demand dial-out capability and tunneling authenticated traffic back to the enterprise is a key differentiator for ZTNA. Also, this functionality can be implemented in part by the application **350** on the user device **300**. Also, the applications **402**, **404** can include B2B applications. Note, the difference between the applications **402**, **404** is the applications **402** are hosted in the cloud, whereas the applications **404** are hosted on the enterprise network **410**. The B2B service described herein contemplates use with either or both of the applications **402**, **404**.

The paradigm of virtual private access systems and methods is to give users network access to get to an application and/or file share, not to the entire network. If a user is not authorized to get the application, the user should not be able even to see that it exists, much less access it. The virtual private access systems and methods provide an approach to deliver secure access by decoupling applications **402**, **404** from the network, instead of providing access with a connector **400**, in front of the applications **402**, **404**, an application on the user device **300**, a central authority **152** to push policy, and the cloud-based system **100** to stitch the applications **402**, **404** and the software connectors **400** together, on a per-user, per-application basis.

With the virtual private access, users can only see the specific applications **402**, **404** allowed by the central authority **152**. Everything else is "invisible" or "dark" to them. Because the virtual private access separates the application from the network, the physical location of the application **402**, **404** becomes irrelevant—if applications **402**, **404** are located in more than one place, the user is automatically directed to the instance that will give them the best performance. The virtual private access also dramatically reduces configuration complexity, such as policies/firewalls in the data centers. Enterprises can, for example, move applications to Amazon Web Services or Microsoft Azure, and take advantage of the elasticity of the cloud, making private, internal applications behave just like the marketing leading enterprise applications. Advantageously, there is no hardware to buy or deploy because the virtual private access is a service offering to end-users and enterprises.

VPN Architecture

FIG. **7** is a network diagram of a VPN architecture **405** for an intelligent, cloud-based global VPN. For illustration purposes, the VPN architecture **405** includes the cloud-based system **100**, the Internet **104**, the applications **402** in SaaS/public cloud systems, and the enterprise network **410**. The VPN architecture **405** also includes a user **102**, which can include any computing device/platform connecting to the cloud-based system **100**, the Internet **104**, the applications **402**, and the enterprise network **410**. The VPN architecture **405** includes a single user **102** for illustration purposes, but those of ordinary skill in the art will recognize that the VPN architecture **405** contemplates a plurality of users **102**. The user **102** can be a nomadic user, a regional/branch office, etc. That is, the user **102** can be any user of the enterprise network **410** that is physically located outside a firewall **412** associated with the enterprise network **410**. The SaaS/public cloud systems can include any systems containing computing and data assets in the cloud such as, for example, Microsoft OneDrive, Google Drive, Dropbox, Apple iCloud, Customer Relationship Management (CRM) systems, SCM, Sales management systems, etc. The enterprise network **410** includes local computing and data assets behind the firewall **412** for additional security on highly confidential assets or legacy assets not yet migrated to the cloud.

The user **102** needs to access the Internet **104**, the SaaS/public cloud systems for the applications **402**, and the enterprise network **410**. Again, conventionally, the solution for secure communication, the user **102** has a VPN connection through the firewall **412** where all data is sent to the enterprise network **410**, including data destined for the Internet **104** or the SaaS/public cloud systems for the applications **402**. Furthermore, this VPN connection dials into the enterprise network **410**. The systems and methods described herein provide the VPN architecture **405**, which provides a secure connection to the enterprise network **410** without bringing all traffic, e.g., traffic for the Internet **104** or the SaaS/public cloud systems, into the enterprise network **410** as well as removing the requirement for the user **102** to dial into the enterprise network **410**.

Instead of the user **102** creating a secure connection through the firewall **412**, the user **102** connects securely to a VPN device **420** located in the cloud-based system **100** through a secure connection **422**. Note, the cloud-based system **100** can include a plurality of VPN devices **420**. The VPN architecture **405** dynamically routes traffic between the user **102** and the Internet **104**, the SaaS/public cloud systems for the applications **402**, and securely with the enterprise network **410**. For secure access to the enterprise network **410**, the VPN architecture **405** includes dynamically creating connections through secure tunnels between three entities: the VPN device **420**, the cloud, and an on-premises redirection proxy **430**. The connection between the cloud-based system **100** and the on-premises redirection proxy **430** is dynamic, on-demand and orchestrated by the cloud-based system **100**. A key feature of the systems and methods is its security at the edge of the cloud-based system **100**—there is no need to punch any holes in the existing on-premises firewall **412**. The on-premises redirection proxy **430** inside the enterprise network **410** "dials out" and connects to the cloud-based system **100** as if too were an end-point via secure connections **440**, **442**. This on-demand dial-out capability and tunneling authenticated traffic back to the enterprise network **410** is a key differentiator.

The VPN architecture **405** includes the VPN devices **420**, the on-premises redirection proxy **430**, a topology controller **450**, and an intelligent DNS proxy **460**. The VPN devices **420** can be Traffic (VPN) distribution servers and can be part of the cloud-based system **100**. In an embodiment, the cloud-based system **100** can be a security cloud such as available from Zscaler, Inc. (www.zscaler.com) performing functions on behalf of every client that connects to it: a) allowing/denying access to specific Internet sites/apps—based on security policy and absence/presence of malware in those sites, and b) set policies on specific SaaS apps and allowing/denying access to specific employees or groups.

The on-premises redirection proxy **430** is located inside a perimeter of the enterprise network **410** (inside the private cloud or inside the corporate data center—depending on the deployment topology). It is connected to a local network and acts as a "bridge" between the users **102** outside the perimeter and apps that are inside the perimeter through the secure connections **440**, **442**. But, this "bridge" is always closed—

it is only open to the users **102** that pass two criteria: a) they must be authenticated by an enterprise authentication service **470**, and b) the security policy in effect allows them access to "cross the bridge."

When the on-premises redirection proxy **430** starts, it establishes a persistent, long-lived connection **472** to the topology controller **450**. The topology controller **450** connects to the on-premises redirection proxy **430** through a secure connection **472** and to the cloud-based system **100** through a secure connection **480**. The on-premises redirection proxy **430** waits for instruction from the topology controller **450** to establish tunnels to specific VPN termination nodes, i.e., the VPN devices **420**, in the cloud-based system **100**. The on-premises redirection proxy **430** is most expediently realized as custom software running inside a virtual machine (VM). The topology controller **450**, as part of the non-volatile data for each enterprise, stores the network topology of a private network of the enterprise network **410**, including, but not limited to, the internal domain name(s), subnet(s) and other routing information.

The DNS proxy **460** handles all domain names to Internet Protocol (IP) Address resolution on behalf of endpoints (clients). These endpoints are user computing devices—such as mobile devices, laptops, tablets, etc. The DNS proxy **460** consults the topology controller **450** to discern packets that must be sent to the Internet **104**, the SaaS/public cloud systems, vs. the enterprise network **410** private network. This decision is made by consulting the topology controller **450** for information about a company's private network and domains. The DNS proxy **460** is connected to the user **102** through a connection **482** and to the cloud-based system **100** through a connection **484**.

The VPN device **420** is located in the cloud-based system **100** and can have multiple points-of-presence around the world. If the cloud-based system **100** is a distributed security cloud, the VPN device **420** can be located with enforcement nodes **150**. In general, the VPN device **420** can be implemented as software instances on the enforcement nodes **150**, as a separate virtual machine on the same physical hardware as the enforcement nodes **150**, or a separate hardware device such as the server **200**, but part of the cloud-based system **100**. The VPN device **420** is the first point of entry for any client wishing to connect to the Internet **104**, SaaS apps, or the enterprise private network. In addition to doing traditional functions of a VPN server, the VPN device **420** works in concert with the topology controller **450** to establish on-demand routes to the on-premises redirection proxy **430**. These routes are set up for each user on demand. When the VPN device **420** determines that a packet from the user **102** is destined for the enterprise private network, it encapsulates the packet and sends it via a tunnel between the VPN device **420** and the on-premises redirection proxy **430**. For packets meant for the Internet **104** or SaaS clouds, the VPN device **420** can forwards it to the enforcement nodes **150**—to continue processing as before or send directly to the Internet **104** or SaaS clouds.

VPN Process

FIG. **8** is a flowchart of a VPN process **500** for an intelligent, cloud-based global VPN. The VPN process **500** can be implemented through the VPN architecture **405**. The VPN process **500** includes the user **102** connecting to the cloud-based system **100** through authentication (step **510**). Once the authentication is complete, a VPN is established between the user **102** and a VPN server in the cloud-based system **100** and DNS for the user **102** is set to a DNS proxy **460** (step **520**). Now, the user **102** has a secure VPN connection to the cloud-based system **100**. Subsequently, the user **102** sends a request to the cloud-based system **100** via the DNS proxy **460** (step **530**). Here, the request can be anything—request for the enterprise network **410**, the Internet **104**, the applications **402** in the SaaS/public cloud systems, the applications **404** in the enterprise network **410**, etc. The DNS proxy **460** contacts the topology controller **450** with the identity of the user and the request (step **540**). That is, whenever the user **102** wishes to reach a destination (Internet, Intranet, SaaS, etc.), it will consult the DNS proxy **460** to obtain the address of the destination.

For non-enterprise requests, the cloud-based system **100** forwards the request per policy (step **550**). Here, the cloud-based system **100** can forward the request based on the policy associated with the enterprise network **410** and the user **102**. With the identity of the user and the enterprise they belong to, the VPN server will contact the topology controller **450** and pre-fetch the enterprise private topology. For enterprise requests, the topology controller **450** fetches a private topology of the enterprise network **410**, instructs the redirection proxy **430** to establish an outbound tunnel to the VPN server, the redirection proxy **430** establishes the outbound tunnel, and requests are forward between the user **102** and the enterprise network **410** securely (step **560**). Here, the DNS proxy **460** works with the topology controller **450** to determine the local access in the enterprise network **410**, and the topology controller **450** works with the redirection proxy **430** to dial out a secure connection to the VPN server. The redirection proxy **430** establishes an on-demand tunnel to the specific VPN server so that it can receive packets meant for its internal network.

Global VPN Applications

Advantageously, the systems and methods avoid the conventional requirement of VPN tunneling all data into the enterprise network **410** and hair-pinning non-enterprise data back out. The systems and methods also allow the enterprise network **410** to have remote offices, etc. without requiring large hardware infrastructures—the cloud-based system **100** bridges the users **102**, remote offices, etc. to the enterprise network **410** in a seamless manner while removing the requirement to bring non-enterprise data through the enterprise network **410**. This recognizes the shift to mobility in enterprise applications. Also, the VPN tunnel on the user **102** can leverage and use existing VPN clients available on the user devices **300**. The cloud-based system **100**, through the VPN architecture **405**, determines how to route traffic for the user **102** efficiently—only enterprise traffic is routed securely to the enterprise network **410**. Additionally, the VPN architecture **405** removes the conventional requirement of tunneling into the enterprise network **410**, which can be an opportunity for security vulnerabilities. Instead, the redirection proxy **430** dials out of the enterprise network **410**.

The systems and methods provide, to the user (enterprise user), a single, seamless way to connect to Public and Private clouds—with no special steps needed to access one vs. the other. To the IT Admin, the systems and methods provide a single point of control and access for all users—security policies and rules are enforced at a single global cloud chokepoint—without impacting user convenience/performance or weakening security.

Virtual Private Access Via the Cloud

FIG. **9** is a network diagram illustrating the cloud-based system **100** with private applications **402**, **404** and data centers **610** connected thereto to provide virtual private access through the cloud-based system **100**. In an aspect, the virtual private access described herein leverages the cloud-based system **100** to enable various users **102** including remote users, contractors, partners, business customers, etc., i.e., anyone who needs access to the private applications **402**, **404** and the data centers **610** access, without granting unfettered access to the internal network, without requiring hardware or appliances, and in a seamless manner from the users' **102** perspective. The private applications **402**, **404** include applications dealing with financial data, personal data, medical data, intellectual property, records, etc., that is the private applications **404** can be available on the enterprise network **410**, but not available remotely except conventionally via VPN access. Examples of the private applications **402**, **404** can include Customer Relationship Management (CRM), sales automation, financial applications, time management, document management, etc. Also, the applications **402**, **404** can be B2B applications or services as described herein.

The virtual private access is a new technique for the users **102** to access the file shares and applications **402**, **404**, without the cost, hassle or security risk of VPNs, which extend network access to deliver app access. The virtual private access decouples private internal applications from the physical network to enable authorized user access to the file shares and applications **402**, **404**, without the security risk or complexity of VPNs. That is, virtual private access takes the "Network" out of VPNs.

In the virtual private access, the users **102**, the file shares and applications **402**, **404**, are communicatively coupled to the cloud-based system **100**, such as via the Internet **104** or the like. On the client-side, at the users **102**, the applications **402**, **404** provision both secure remote access and optionally accessibility to the cloud-based system **100**. The application **402**, **404** establishes a connection to the closest enforcement node **150** in the cloud-based system **100** at startup and may not accept incoming requests.

At the file shares and applications **402**, **404**, the lightweight connectors **400** sit in front of the applications **402**, **404**. The lightweight connectors **400** become the path to the file shares and applications **402**, **404** behind it, and connect only to the cloud-based system **100**. The lightweight connectors **400** can be lightweight, ephemeral binary, such as deployed as a virtual machine, to establish a connection between the file shares and applications **402**, **404** and the cloud-based system **100**, such as via the closest enforcement node **150**. The lightweight connectors **400** do not accept inbound connections of any kind, dramatically reducing the overall threat surface. The lightweight connectors **400** can be enabled on a standard VMware platform; additional lightweight connectors **400** can be created in less than 5 seconds to handle additional application instances. By not accepting inbound connections, the lightweight connectors **400** make the file shares and applications **402**, **404** "dark," removing a significant threat vector.

The policy can be established and pushed by policy engines in the central authority **152**, such as via a distributed cluster of multi-tenant policy engines that provide a single interface for all policy creation. Also, no data of any kind transits the policy engines. The enforcement nodes **150** in the security cloud stitch connections together, between the users **102** and the file shares and applications **402**, **404**, without processing traffic of any kind. When the user **102** requests an application in the file shares and applications **402**, **404**, the policy engine delivers connection information to the application **350** and app-side enforcement nodes **150**, which includes the location of a single enforcement nodes **150** to provision the client/app connection. The connection is established through the enforcement nodes **150**, and is encrypted with a combination of the customer's client and server-side certificates. While the enforcement nodes **150** provision the connection, they do not participate in the key exchange, nor do they have visibility into the traffic flows.

Advantageously, the virtual private access provides increased security in that the file shares and applications **402**, **404** are visible only to the users **102** that are authorized to access them; unauthorized users are not able to even see them. Because application access is provisioned through the cloud-based system **100**, rather than via a network connection, the virtual private access makes it impossible to route back to applications. The virtual private access is enabled using the application **350**, without the need to launch or exit VPN clients. The application access just works in the background enabling application-specific access to individual contractors, business partners or other companies, i.e., the users **102**.

The virtual private access provides capital expense (CAPEX) and operating expense (OPEX) reductions as there is no hardware to deploy, configure, or maintain. Legacy VPNs can be phased out. Internal IT can be devoted to enabling business strategy, rather than maintaining network "plumbing." Enterprises can move apps to the cloud on their schedule, without the need to re-architect, set up site-to-site VPNs or deliver a substandard user experience.

The virtual private access provides easy deployment, i.e., put lightweight connectors **400** in front of the file shares and applications **402**, **404**, wherever they are. The virtual private access will automatically route to the location that delivers the best performance. Wildcard app deployment will discover applications upon request, regardless of their location, then build granular user access policies around them. There is no need for complex firewall rules, Network Address Translation issues or policy juggling to deliver application access. Further, the virtual private access provides seamless integration with existing Single Sign-On (SSO) infrastructure.

FIG. **10** is a network diagram of a virtual private access network **700**A and a flowchart of a virtual private access process **750** implemented thereon. The cloud-based system **100** includes three enforcement nodes **150**A, **150**B, **150**C, assume for illustration purposes in San Francisco, New York, and London, respectively. The user **102** has the application **350** executing on the user device **300**, which is communicatively coupled to the enforcement node **150**A. The enterprise file share and application **402**, **404** is communicatively coupled to the enforcement node **150**C. Note, there can be direct connectivity between the enforcement nodes **150**A, **150**C, the enforcement nodes **150**A, **150**C can connect through the enforcement node **150**B, or both the user **102** and the enterprise file share and application **402**, **404** can be connected to the same node **150**. That is, the architecture of the cloud-based system **100** can include various implementations.

The virtual private access process **750** is described with reference to both the user **102**, the cloud-based system **100**, and the enterprise file share and application **402**, **404**. First, the user **102** is executing the application **350** on the user device **300**, in the background. The user **102** launches the application **350** and can be redirected to an enterprise ID provider or the like to sign on, i.e., a single sign on, without setting up new accounts. Once authenticated, Public Key Infrastructure (PKI) certificate **720** enrollment occurs, between the user **102** and the enforcement node **150**A. With the application **350** executing on the user device, the user **102** makes a request to the enterprise file share and application **402**, **404**, e.g., intranet.company.com, crm.company-.com, etc. (step **752**). Note, the request is not limited to web applications and can include anything such as a remote desktop or anything handling any static Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) applications.

This request is intercepted by the enforcement node **150**A and redirected to the central authority **152**, which performs a policy lookup for the user **102** and the user device **300** (step **754**), transparent to the user **102**. The central authority **152** determines if the user **102** and the user device **300** are authorized for the enterprise file share and application **402**, **404**. Once authorization is determined, the central authority **152** provides information to the enforcement nodes **150**A, **150**B, **150**C, the application **350**, and the lightweight connectors **400** at the enterprise file share and application **402**, **404**, and the information can include the certificates **720** and other details necessary to stitch secure connections between the various devices. Specifically, the central authority **152** can create connection information with the best enforcement nodes **150** for joint connections, from the user **102** to the enterprise file share and application **402**, **404**, and the unique tokens (step **756**). With the connection information, the enforcement node **150**A connects to the user **102**, presenting a token, and the enforcement node **150**C connects to the lightweight connector **400**, presenting a token (step **758**). Now, a connection is stitched between the user **102** to the enterprise file share and application **402**, **404**, through the application **350**, the enforcement nodes **150**A, **150**B, **150**C, and the lightweight connector **400**.

Comparison—VPN with Virtual Private Access

FIGS. **11** and **12** are network diagrams of a VPN configuration (FIG. **11**) compared to virtual private access (FIG. **12**) illustrating the differences therein. In FIG. **11**, a user device **300** connects to a VPN termination device **804** associated with an enterprise network **806** via the Internet **104**, such that the user device **300** is on the enterprise network **806**, where associated applications reside. Of course, any malware on the user device **300** or anyone that steals the user device **300** is also on the enterprise network **806**. The VPN termination device **804** creates a Distributed Denial-of-Service (DDoS) attack surface, adds infrastructure cost and creates network complexity as applications grow. Conversely, in FIG. **12**, the user device **300** uses the virtual private access via the cloud-based system **100** to connect to the lightweight connector **400** associated with a specific application **404**. The virtual private access provides granular access by the user device **300** and the application, and the user device **300** is not on the enterprise network **806**. Thus, the application is never directly exposed to the user device **300**, the security cloud handles provisioning, and the traffic remains completely private.

Comparison—Private Applications in the Public Cloud

FIGS. **13** and **14** are network diagrams of conventional private application access in the public cloud (FIG. **13**) compared to private application in the public cloud with virtual private access (FIG. **14**). In FIG. **13**, the user device **300** still has to connect to the enterprise network **806** via the VPN termination device **804** as in FIG. **11**, and the cloud applications, such as in the data center **610**, are accessible via the enterprise network **806** via a site-to-site VPN between the enterprise network **806** and the data center **610**. Disadvantageously, the user experience is eroded for the user device **300** and agility is hampered for the enterprise by networking concerns and capability. In FIG. **14**, the virtual private access abstracts the application **402**, in the data center **610**, from the IP address, so location is irrelevant. The enterprise can move private applications to the cloud securely, as needed.

Comparison—Contractor/Private Application Access

FIGS. **15** and **16** are network diagrams of conventional contractor/partner access (FIG. **15**) of applications in the enterprise network **806** compared to contractor/partner access (FIG. **16**) of the applications with virtual private access. Contractor/partner access includes providing third parties access to applications on the enterprise network **806**, for a variety of purposes. In FIG. **15**, similar to FIGS. **11** and **13**, contractor/partner access includes VPN connections to the VPN termination device **804**, providing contractor/partners **820** full access to the enterprise network **806**, not just the specific application or asset that they require. Unfortunately, stolen credentials can allow hackers to get into networks or to map assets for later assault. In FIG. **16**, the virtual private access, using the cloud-based system **100**, allows access specific to applications or assets as needed by the contractor/partners **820**, via the lightweight connector **400**. Thus, the contractor/partners **820** do not have full network access, the access is specific to each user, and the connections are provisioned dynamically, avoiding a direct network connection that can be misused or exploited.

Comparison—Example Application—M&A Data Access

FIGS. **17** and **18** are network diagrams of a conventional network setup to share data between two companies (FIG. **17**) such as for Merger and Acquisition (M&A) purposes or the like, compared to a network setup using virtual private access (FIG. **18**). Conventionally, the two companies provide VPN connections between their associated enterprise networks **806**A, **806**B to one another. Each company gets "all or nothing"—no per-application granularity. Disadvantageously, creating Access Control Lists (ACLs)/firewall rules and NATting through each companies' respective firewalls is very complex, particularly with overlapping internal IP addressing. In FIG. **18**, the virtual private access allows connections provisioned by the user and device to the application by name, not by IP address, authorized users can access only specific applications, not an entire network, and firewall complexities disappear.

Administrative View of Virtual Private Access

FIGS. **19** and **20** are screenshots of Graphical User Interfaces (GUIs) for administrator access to the virtual private access. FIG. **19** illustrates a GUI of network auto-discovery and FIG. **20** illustrates a GUI for reporting. For network and application discovery, the virtual private access can use wildcard application discovery where a Domain/name-based query to the lightweight connector **400** will show company applications behind them. This allows the discovery of internal applications as users request them using "*.company.com" to find applications. Then, the granular policy can be built around the applications to dramatically simply startup. Further, the virtual private access can show the location of users that are accessing private/internal applications, including identifying anomalous access patterns to assist in stopping possible data leakage or compliance violation.

Virtual Private Access

In an embodiment, a virtual private access method implemented by a cloud-based system, includes receiving a request to access resources from a user device, wherein the resources are located in one of a public cloud and an enterprise network and the user device is remote therefrom on the Internet; forwarding the request to a central authority for a policy look up and for a determination of connection information to make an associated secure connection through the cloud-based system to the resources; receiving the connection information from the central authority responsive to an authorized policy look up; and creating secure tunnels between the user device and the resources based on the connection information. Prior to the receiving, a user executes an application on the user device, provides authentication, and provides the request with the application operating on the user device. The application can be configured to connect the user device to the cloud-based system, via an optimized cloud node based on a location of the user device. The resources can be communicatively coupled to a lightweight connector operating on a computer and communicatively coupled between the resources and the cloud-based system. The virtual private access method can further include detecting the resources based on a query to the lightweight connector. The lightweight connector can be prevented from accepting inbound connections, thereby preventing access of the resources external from the public cloud or the enterprise network. The creating secure tunnels can include creating connections between one or more cloud nodes in the cloud-based system, wherein the one or more cloud nodes do not participate in a key exchange, and the one or more cloud nodes do not have data access to traffic on the secure tunnels. The creating secure tunnels can include creating connections between one or more cloud nodes in the cloud-based system, wherein the one or more cloud nodes create the secure tunnels based on a combination of a client-side certificate and a server-side certificate. The secure tunnels can be created through software on the user device, the cloud-based system, and a lightweight connector operating on a computer associated with the resources, thereby eliminating dedicated hardware for virtual private network connections.

In another embodiment, a cloud-based system adapted to implement virtual private access includes one or more cloud nodes communicatively coupled to one another; wherein each of the one or more cloud nodes includes one or more processors and memory storing instructions that, when executed, cause the one or more processors to receive a request to access resources from a user device, wherein the resources are located in one of a public cloud and an enterprise network and the user device is remote therefrom on the Internet; forward the request to a central authority for a policy look up and for a determination of connection information to make an associated secure connection through the cloud-based system to the resources; receive the connection information from the central authority responsive to an authorized policy look up; and create secure tunnels between the user device and the resources based on the connection information. Prior to reception of the request, a user executes an application on the user device, provides authentication, and provides the request with the application operating on the user device. The application can be configured to connect the user device to the cloud based system, via an optimized cloud node based on a location of the user device. The resources can be communicatively coupled to a lightweight connector operating on a computer and communicatively coupled between the resources and the cloud-based system. The memory storing instructions that, when executed, can further cause the one or more processors to detect the resources based on a query to the lightweight connector. The lightweight connector can be prevented from accepting inbound connections, thereby preventing access of the resources external from the public cloud or the enterprise network. The secure tunnels can be created through connections between one or more cloud nodes in the cloud-based system, wherein the one or more cloud nodes do not participate in a key exchange, and the one or more cloud nodes do not have data access to traffic on the secure tunnels. The secure tunnels can be created through connections between one or more cloud nodes in the cloud-based system, wherein the one or more cloud nodes create the secure tunnels based on a combination of a client-side certificate and a server-side certificate. The secure tunnels can be created through software on the user device, the cloud-based system, and a lightweight connector operating on a computer associated with the resources, thereby eliminating dedicated hardware for virtual private network connections.

Software stored in a non-transitory computer readable medium including instructions executable by a system, which in response to such execution causes the system to perform operations including receiving a request to access resources from a user device, wherein the resources are located in one of a public cloud and an enterprise network and the user device is remote therefrom on the Internet; forwarding the request to a central authority for a policy look up and for a determination of connection information to make an associated secure connection through the cloud-based system to the resources; receiving the connection information from the central authority responsive to an authorized policy look up; and creating secure tunnels between the user device and the resources based on the connection information. The resources can be communicatively coupled to a lightweight connector operating on a computer and communicatively coupled between the resources and the cloud-based system, and wherein the instructions executable by the system, which in response to such execution can further cause the system to perform operations including detecting the resources based on a query to the lightweight connector.

VPN in the Cloud

In an embodiment, a method includes connecting to a client at a Virtual Private Network (VPN) device in a cloud-based system; forwarding requests from the client for the Internet or public clouds accordingly; and for requests for an enterprise associated with the client, contacting a topology controller to fetch a topology of the enterprise, causing a tunnel to be established from the enterprise to the VPN device, and forwarding the requests for the enterprise through the tunnel to the cloud-based system for proactive monitoring; and providing a secure connection from the cloud-based system back to the enterprise, including internal domain and subnets associated with the enterprise. The method can further include authenticating, via an authentication server, the client prior to the connecting and associated the client with the enterprise. The method can further include, subsequent to the connecting, setting a Domain Name Server (DNS) associated with the cloud-based system to provide DNS lookups for the client. The method can further include utilizing the DNS to determine a destination of the requests; and for the requests for the enterprise, contacting the topology controller to pre-fetch the topology of the enterprise. The method can further include operating an on-premises redirection proxy within the enterprise, wherein the on-premises redirection proxy is configured to establish the tunnel from the enterprise to the VPN device. Secure tunnels to the enterprise are dialed out from the enterprise by the on-premises redirection proxy. The on-premises redirection proxy is a virtual machine operating behind a firewall associated with the enterprise. The on-premises redirection proxy is configured as a bridge between the client and applications inside the enterprise. The VPN device operates on a cloud node in the cloud-based system, and wherein the cloud-based system includes a distributed security cloud. The VPN device can include one of a software instance on a cloud node or a virtual machine on the cloud node. The topology controller includes a network topology of the enterprise, including internal domain names and subnets.

In another embodiment, a cloud-based system includes one or more Virtual Private Network (VPN) servers, wherein one or more clients connect securely to the one or more VPN servers; a topology controller communicatively coupled to the one or more VPN servers; a Domain Name Server (DNS) communicatively coupled to the topology controller and the one or more VPN servers; and a redirection proxy located in a private network and communicatively coupled to the one or more VPN servers and the topology controller; wherein requests from the one or more clients to the private network cause on demand secure connections being established by the redirection proxy to associated VPN servers in a cloud-based system, wherein the on demand secure connections provide connectivity to the private network including internal domain and subnets associated with the private network, and wherein the cloud-based system performs proactive monitoring. Requests from the one or more clients outside of the private network are forwarded without traversing the private network. The redirection proxy maintains a persistent connection to the topology controller and establishes secure tunnels to the one or more VPN servers based on direction from the topology controller. The topology controller includes a network topology of the private network, including internal domain names and subnets. The VPN servers operate on cloud nodes in a distributed security cloud.

In yet another embodiment, a VPN system includes a network interface, a data store, and a processor, each communicatively coupled together; and memory storing instructions that, when executed, cause the processor to: establish a secure tunnel with a client; forward requests from the client to the Internet accordingly; and for requests to an enterprise, contact a topology controller to fetch a topology of the enterprise, cause a tunnel to be established from the enterprise to the VPN system, and forwarding the requests for the enterprise through the tunnel and the secure tunnel, wherein the secure tunnel is achieved by using an on-demand dial-out and tunneling traffic authentication. The memory storing instructions that, when executed, further cause the processor to: cause the tunnel to be established from the enterprise to the VPN system through an on premises redirection proxy located within the enterprise.

Clientless Connection Setup

In an embodiment, the user device **300** can include the application **350** for performing the various virtual private access processes described herein. An example of such a client app is described in commonly-assigned U.S. Pat. No. 9,935,955, issued Apr. 3, 2018, and entitled "SYSTEMS AND METHODS FOR CLOUD BASED UNIFIED SERVICE DISCOVERY AND SECURE AVAILABILITY," the contents of which are incorporated by reference.

In another embodiment, the user **102** can connect via virtual private access in a so-called clientless connection approach described herein. The clientless connection approach can use a standard Web browser on the user device **300**. The clientless connection approach can be used with third-party users **102** (e.g., contractors), for one-time use by a user **102**, or in situations where it is simply undesired to install the application **350**. The object of the clientless connection approach is to perform the same techniques described herein for virtual private access, without the application **350** (alternatively, the application **350** can be viewed as the standard Web browser in the clientless approach).

FIG. **21** is a block diagram of logical components of a clientless connection approach **900**. FIG. **22** is a block diagram of the logical components of the clientless connection approach **900** illustrating the steps for configuration. FIG. **23** is a block diagram of the logical components of the clientless connection approach **900** illustrating the steps for a connection setup subsequent to the configuration.

The logical components are generalized at a user device **300**, centralized components **904**, and distributed components **906**. The user device **300** can be the user **102** or any of the other users described herein. This can be a user device **300** or any other type of device which can execute a Web browser. The centralized components **904** can be compute and storage resources in the cloud, on dedicated servers **200**, or the like. For example, the centralized components **904** can be implemented in Amazon Web Services (AWS) infrastructure. Other embodiments are also contemplated. The distributed components **906** can be through the cloud-based system **100**, such as through the enforcement nodes **150**. The user device **300** can include a User Interface (UI) **910** and a Web browser **912**. The UI **910** can be operated through the Web browser **912**. The Web browser **912** can be a standard Web browser such as Google Chrome, Microsoft Edge/Explorer, Mozilla Firefox, Apple Safari, etc.

The centralized components **904** include an Application Programming Interface (API) **914**, a database (DB) **916**, a crypto service **918**, a cookie store **920**, and Security Assertion Markup Language (SAML) Service Provider (SP) component **922**. The API **914** enables connectivity to the user device **300** for purposes of configuration. The database **916** stores public and private keys. The crypto service **918** creates/provides private keys. The SAML SP component **922** is used during the sign on process. The distributed components **906** includes a broker **924** and an exporter **926** for creating clientless connections with the user device **300**. The distributed components **906** can be implemented at each of the enforcement nodes **150** in the cloud-based system **100**.

In FIG. **21**, a user device **300** performs a configuration such as through the UI **910**, where a user of the user device **300** uploads a private+public key (step C**1**). Of note, the user can use existing private+public keys, create private+public keys through various providers, obtain a private+public key from the cloud-based system **100**, etc. These keys are available at the user device **300** and provided to the cloud-based system **100** via the API **914** through the step C**1**.

The API **914** requests the crypto service **918** encrypt the private key (step C**2**). This request can include a namespace for the encryption context, e.g., the request can indicate 'for exporter' or something similar to designate the use in the clientless virtual private access, conceivably even per customer. The crypto service **918** encrypts the private key and returns the encrypted private key to the API **914** (step C**3**). The API **914** stores the public+private key in the database **916** (step C**4**). Here, the private key is encrypted, whereas the public key is not.

In FIG. **22**, a user device **300** is already configured with a private+public key as described in FIG. **21** and performs a clientless connection via the Web browser **912** to an address, e.g., app.cust.com. The Web browser **912** can request app.cust.com from a DNS server **930**, which CNAME's app.cust.com to an address of the exporter **926**, e.g., cust.export.zscaler.com, which can resolve to the nearest enforcement node **150** acting as the exporter **926** (step S**1**).

The Web browser **912** can set a GET request to the exporter **926** using Transmission Control Protocol (TCP) port 80 (the most common port for HTTP), and the exporter **926** recognizes the hostname as app.cust.com and redirects to TCP port **443** (which is used for Secure Sockets Layer (SSL)) (step S**2**).

The Web browser **912** opens Transport Layer Security (TLS) to the exporter **926** and sends the GET request (step S**3**). The exporter **926** utilizes the Server Name Indication (SNI) to determine the certificate to present. SNI is an extension to the TLS protocol by which the user device **300** indicates which hostname it is attempting to connect to at the start of the handshaking process. The exporter can obtain the encrypted certificate from the database **916**, and the certificate decryption key from the crypto service **918** and the crypto service **918** authenticates the exporter **926** based on the certificate credentials (step S**4**).

The exporter **926** parses HTTP and determines there is no valid cookie with the GET request (step S**5**). Note, if there were a valid cookie, the process could go to step S**12**. Note, other HTTP methods are also contemplated besides the GET request. The exporter **926** can redirect to the SAML SP component **922**, encoding the original GET request in a query string.

The SAML SP component **922** performs normal SAML functions and redirects to an Identity Provider (IDP) **932** (step S**6**). The IDP **932** authenticates the user device **300** and redirects back to the SAML SP component **922** (step S**7**). The SAML SP component **922** receives an authentication response (assertion) (step S**8**).

The SAML SP component **922** generates a key (cookie), which is random, plus some coding of the cookie store **920** in which it will be stored (step S**9**). The cookie and assertion are stored in the cookie store **920**. The SAML SP component **922** redirects the user device **300** back to app.cust.com/zscalerauth/. . . , encoding the original GET request+cookie in the query string (step S**10**). Specifically, the redirect address includes app.cust.com plus a designation of authentication (/zscalerauth/) plus the original GET request and cookie in the encoded query string.

The exporter **926** receives the request for app.cust.com/zscalerauth/. . . with an encoded query string (step S**11**), and the exporter **926** redirects back to the original URL (app.cust.com), including a set-cookie with the authentication cookie encoded in steps S**9**, S**10**.

The exporter **926** receives the request with the authentication cookie (step S**12**). The exporter **926** fetches the assertion from the cookie store **920** (step S**13**). If the cookie (assertion) does not exist, is expired, etc., then the process returns to step S**5**.

The exporter **926** creates a virtual private access client context and authenticates to a broker **924** using end user assertion (step S**14**). The exporter **926** opens (or reuses an idle) tunnel for a request to app.cust.com. (step S**15**). As the response arrives from the tunnel, it is sent back to the Web browser **912**. Thus, the exporter **926** makes the virtual private access connection between the user device **300** through the Web browser **912** and app.cust.com.

Secure Access for B2B Applications

FIG. **24** is a network diagram of a secure access for B2B applications network **1000**, including the cloud-based system **100** interconnecting users **102**A, **102**B with the applications **402**, **404**. Again, the present disclosure includes secure access for B2B applications. Again, the B2B applications can include business-facing applications such as Supply Chain Management (SCM) applications, inventory management applications, ordering applications, financial applications, payroll applications, etc. For example, B2B applications can be web-based applications, i.e., the applications **402**, **404**, hosted either in the enterprise network **410**, in the public cloud, in the data center **610**, etc. The secure access for B2B applications leverages the various elements, architecture, and functionality described herein. Specifically, the secure access for B2B applications is a cloud service such as through the cloud-based system **100** with the enforcement nodes **150** and the central authority **152**, for fast, seamless, scalable, and reliable access to B2B resources. Functionally, the secure access provides policy-based access for all B2B apps, regardless of hosted application location, B2B users **102** receive seamless access to the applications **402**, **404** in the data center **610**, in the enterprise network **410**, in hybrid environments, etc.

The secure access includes multiple identity provider support, enabling simultaneous support of multiple IDP services from B2B identities. The secure access includes a B2B user portal **1002** to provides B2B users **102**A, **102**B (e.g., suppliers A, B) with a unique portal that delivers a consolidated display of all authorized B2B applications available for access. The B2B user portal **1002** can be a GUI that is provided to the B2B users **102**A, **102**B, and may be implemented via the cloud-based system **100** or some other server **200**. The objective of the secure access is to provide administrators global visibility of B2B users **102** and applications **402**, **404** with a centralized control plane logging diagnostics for all B2B activity in the past and in real-time. The secure access can include a B2B log streaming service with support for automatic streaming of all user, app, and connector logs to SIEM.

The secure access includes browser access traffic forwarding enabling secure B2B access to all web applications **402**, **404** with just a simple browser. No client deployment on the device is required, such as described herein in the clientless approach. The secure access for B2B applications network **1000** includes B2B brokers **1004** and the connectors **400**. The B2B brokers **1004** can include the enforcement nodes **150** and are configured to enforce B2B access policies and stitch together an end-to-end tunnel for app-to-business user connection in a location closest to the user **102**A, **102**B for fast access. As described herein, the connectors **400** are lightweight, such as VM-based, in the data center, cloud, and hybrid environments to enable secure connectivity to B2B applications **402**, **404**. Again, as described herein, the connectors **400** provides inside-out connections to the B2B broker **1004** to ensure the applications **402**, **404** are never exposed.

The secure access for B2B applications network **1000** is illustrated with two users **102**A, **102**B (suppliers A, B) and with two example applications **402**, **404** for illustration purposes. Of course, additional users **102** and applications **402**, **404** are contemplated. The flow in the secure access for B2B is described herein with reference to the secure access for B2B applications network **1000**.

A user **102**A, **102**B requests a URL and is redirected to a corresponding identity provider (IDP) (step A**1**). For example, the identity provider could include Okta, Microsoft, etc. For example, the URL can be apps.myacme.com, and the request from the user **102**A, **102**B is received in the cloud-based system **100** (e.g., based on inline monitoring), and the cloud-based system **100** can redirect (using Canonical Name record (CNAME)) to apps.myacme.cloud.com for the B2B user portal **1002**. Based on the authentication/authorization through the identity provider, the B2B user portal **1002** is configured to display the applications **402**, **404** available to the authorized user **102**A, **102**B (step A**2**).

Based on the user **102**A, **102**B selecting one of the applications **402**, **404** that are available, the connectors **400** closest to the requested applications **402**, **404** create an inside-out encrypted tunnel (step A**3**). The tunnel is referred to as inside-out because the connector **400** (the endpoint) creates the tunnel to the cloud-based system **100** (i.e., dial-out versus dial-in). In an embodiment, the encrypted tunnel is a TLS (e.g., TLS 1.2) encrypted tunnel over port **443**. The broker **1004** is configured to stitch together the tunnel from the connector **400** to the user **102**A, **102**B (step A**4**).

Further, the cloud-based system **100** provides real-time, global visibility into all user **102** and application **402**, **404** activity, such as via the storage cluster **156** (step A**5**). This unified global visibility enables various reports with visualizations for a customer-centric view. This enables reporting at the level of a B2B customer's usage statistics (and not the individual users within that customer). B2B applications have a significant business impact, and these are accessed by external users (customers). In an embodiment, the present disclosure includes the ability to monitor usage patterns and alert on deviating behavior, for both security and operational reasons. That is various machine learning techniques can be implemented on the log data in the storage cluster **156** for detecting deviating behavior.

FIG. **25** is a network diagram of the secure access for B2B applications network **1000**, illustrating a unique approach to ZTNA, removing exposure to the Internet. With the secure access, the applications **402**, **404** are never exposed to the Internet, the users **102** are never brought on a corporate or enterprise networks, and application segmentation is achieved without network segmentation. The inside-out tunnel and stitched connections are such that the applications **402**, **404** only communicate with the cloud-based system **100**, via the connector **400**, and the users **102** only communicate with the cloud-based system **100**. The applications **402**, **404** are invisible except to authorized business customers. The application segmentation is achieved throughs user to application micro tunnels.

FIG. **26** is a network diagram of the secure access for B2B applications network **1000**, illustrating scalability and agility. The cloud-based system **100** supports scale, and, in an embodiment, the cloud-based system **100** offers a cloud security service that ensures scalability across multiple data centers. The secure access eases migration, so the applications **402**, **404** can be moved across clouds seamlessly and with no static IP.

FIG. **27** is a network diagram of the secure access for B2B applications network **1000** illustrating modernizing legacy applications. Legacy applications **402**, **404** can operate with HTTPS over the Internet **104**. These legacy applications can be modernized with a Single-Sign-On (SSO) experience.

Secure Access to B2B Application Advantages

The secure access provides a zero attack surface for B2B apps, and bad actors cannot attack what they cannot see. This allows business customers to access B2B applications without exposing those services to potential attacks. The access is based on identity and granular access policies set by the application administrator. The B2B applications are made invisible to all unauthorized users, minimizing the attack surface. The data path is fully encrypted end to end, so data privacy and integrity remain intact.

The secure access eliminates administrative overhead, bringing the enterprise's own identity to the public cloud. There is no need to manage the identities for business customers or deal with infrastructure changes moving to the cloud. There is support for multiple identity providers removing the complexity of managing IDPs for each business partner. There is reduced cost incurred from storing business partner licenses. The users **102** see a better experience as there is less need to reauthenticate, and there is a reduction in network and infrastructure changes during the move to public cloud.

The secure access benefits from the scalability of a cloud-delivered service, offloading the costs of managing appliances and achieve the scale that business needs. The 100% cloud-delivered architecture removes the need to manage network appliances altogether. The secure access supports access to B2B apps running in any environment: data center, public cloud, and hybrid cloud. Business customers get direct-to-cloud connectivity and are never backhauled through a central chokepoint.

Finally, the user experience is improved for business customers with a scalable cloud service. The secure cloud access service ensures better user experience for business customers, higher availability of services, and lower network expenses. The brokers **1004** run globally and ensure users access B2B services via the optimal data path There are no network security appliances so network costs are reduced drastically.

Browser Isolation

Browser (web) isolation is a technique where a user's browser or apps are physically isolated away from the user device, the local network, etc. thereby removing the risks of malicious code, malware, cyberattacks, etc. This has been shown to be an effective technique for enterprises to reduce attacks. Techniques for browser isolation are described in commonly-assigned U.S. patent application Ser. No. 16/702,889, filed Dec. 4, 2019, and entitled "Cloud-based web content processing system providing client threat isolation and data integrity," the contents of which are incorporated by reference herein. Traditionally browser isolation was focused of removing the risks of malicious code, malware, cyberattacks, etc. U.S. patent application Ser. No. 16/702,889 describes an additional use case of preventing data exfiltration. That is, because no data is delivered to the local system (e.g., to be processed by web content through the local web browser), none of the confidential or otherwise sensitive data can be retained on the local system.

The secure access can interoperate with browser isolation through the cloud-based system **100**, to prevent data exfiltration, which is extremely critical as this is customer-facing data which adds to the sensitivity and liability, and also accessible to external users (customers). This functionality forces customers to interact with the B2B applications via an isolated, contained environment.

It will be appreciated that some embodiments described herein may include one or more generic or specialized processors ("one or more processors") such as microprocessors; Central Processing Units (CPUs); Digital Signal Processors (DSPs): customized processors such as Network Processors (NPs) or Network Processing Units (NPUs), Graphics Processing Units (GPUs), or the like; Field Programmable Gate Arrays (FPGAs); and the like along with unique stored program instructions (including both software and firmware) for control thereof to implement, in conjunction with certain non-processor circuits, some, most, or all of the functions of the methods and/or systems described herein. Alternatively, some or all functions may be implemented by a state machine that has no stored program instructions, or in one or more Application Specific Integrated Circuits (ASICs), in which each function or some combinations of certain of the functions are implemented as custom logic or circuitry. Of course, a combination of the aforementioned approaches may be used. For some of the embodiments described herein, a corresponding device such as hardware, software, firmware, and a combination thereof can be referred to as "circuitry configured or adapted to," "logic configured or adapted to," etc. perform a set of operations, steps, methods, processes, algorithms, functions, techniques, etc. as described herein for the various embodiments.

Moreover, some embodiments may include a non-transitory computer-readable storage medium having computer readable code stored thereon for programming a computer, server, appliance, device, processor, circuit, etc. each of which may include a processor to perform functions as described and claimed herein. Examples of such computer-readable storage mediums include, but are not limited to, a hard disk, an optical storage device, a magnetic storage device, a ROM (Read Only Memory), a PROM (Programmable Read Only Memory), an EPROM (Erasable Programmable Read Only Memory), an EEPROM (Electrically Erasable Programmable Read Only Memory), Flash memory, and the like. When stored in the non-transitory computer readable medium, software can include instructions executable by a processor or device (e.g., any type of programmable circuitry or logic) that, in response to such execution, cause a processor or the device to perform a set of operations, steps, methods, processes, algorithms, functions, techniques, etc. as described herein for the various embodiments.

Although the present disclosure has been illustrated and described herein with reference to preferred embodiments and specific examples thereof, it will be readily apparent to those of ordinary skill in the art that other embodiments and examples may perform similar functions and/or achieve like results. All such equivalent embodiments and examples are within the spirit and scope of the present disclosure, are contemplated thereby, and are intended to be covered by the following claims.

What is claimed is:

1. A method comprising:

responsive to a request from a user for one or more Business-to-Business (B2B) applications, redirecting the request, by a cloud-based system, to an identity provider to authorize the user;

displaying, via a unique user portal, the one or more B2B applications that the user is authorized to access based on a response from the identity provider, wherein any applications of the one or more B2B applications that the user is unauthorized to access based on the response from the identity provider, are omitted from the unique user portal, such that B2B applications that the user is unauthorized to access are invisible to the user;

responsive to a selection of a B2B application of the one or more B2B applications, creating a first tunnel from the B2B application to the cloud-based system, wherein the tunnel is an inside-out encrypted tunnel, and wherein responsive to the B2B application being located in more than one place, the first tunnel is created from an instance of the B2B application that will give the best performance; and

stitching the first tunnel between the B2B application and the cloud-based system with a second tunnel between the user and the cloud-based system via an enforcement node, wherein the enforcement node has no visibility into traffic between the user and the B2B application, and wherein the B2B application and the user can only communicate with the cloud-based system, and application segmentation is achieved without network segmentation through user to application micro tunnels.

2. The method of claim **1**, further comprising

logging real-time and historical activity and diagnostics of one or more users with the one or more B2B applications and storing the activity with a plurality of users associated with the one or more B2B applications.

3. The method of claim **2**, further comprising

providing a Graphical User Interface (GUI) including visualizations related to real-time and historical logs associated with user transactions with the one or more B2B applications.

4. The method of claim **2**, further comprising

analyzing activity associated with a B2B customer to detect usage patterns; and

providing an alert responsive to any deviating behavior, for both security and operational reasons, wherein the deviating behavior is detected via one or more machine learning techniques.

5. The method of claim **1**, wherein the one or more B2B applications include business facing applications including any of Supply Chain Management (SCM) applications, inventory management applications, ordering applications, financial applications, and payroll applications.

6. The method of claim **1**, wherein the one or more B2B applications are web-based applications and the request is a Uniform Resource Locator (URL).

7. The method of claim **1**, wherein the user provides the request via a web browser executed on a user device.

8. A non-transitory computer-readable medium comprising instructions that, when executed, cause a cloud-based system to perform the steps of:

responsive to a request from a user for one or more Business-to-Business (B2B) applications, redirecting the request to an identity provider to authorize the user;

displaying, via a unique user portal, the one or more B2B applications that the user is authorized to access based on a response from the identity provider, wherein any applications of the one or more B2B applications that the user is unauthorized to access based on the response from the identity provider, are omitted from the unique user portal, such that B2B applications that the user is unauthorized to access are invisible to the user; and

responsive to a selection of a B2B application of the one or more B2B applications and responsive to creation of a first tunnel from the B2B application to the cloud-based system, wherein the tunnel is an inside-out encrypted tunnel, stitching the first tunnel between the B2B application and the cloud-based system with a second tunnel between the user and the cloud-based system via an enforcement node, wherein the enforcement node has no visibility into traffic between the user and the B2B application, and wherein the B2B application and the user can only communicate with the cloud-based system, and application segmentation is achieved without network segmentation through user to application micro tunnels.

9. The non-transitory computer-readable medium of claim **8**, wherein the steps further include

Logging real-time and historical activity and diagnostics of one or more users with the one or more B2B applications and storing the activity with a plurality of users associated with the one or more B2B applications.

10. The non-transitory computer-readable medium of claim **8**, wherein the one or more B2B applications include business facing applications including any of Supply Chain

Management (SCM) applications, inventory management applications, ordering applications, financial applications, and payroll applications.

11. The non-transitory computer-readable medium of claim **8**, wherein the one or more B2B applications are web-based applications and the request is a Uniform Resource Locator (URL).

12. The non-transitory computer-readable medium of claim **8**, wherein the user provides the request via a web browser executed on a user device.

13. A cloud-based system comprising:

a plurality of enforcement nodes interconnected to one another; and

a central authority interconnected to the plurality of enforcement nodes,

wherein any of the plurality of enforcement nodes are connected to one or more Business-to-Business (B2B) applications, via corresponding connectors, and to a user;

wherein the cloud-based system is configured to

responsive to a request from the user for the one or more Business-to-Business (B2B) applications, redirect the request, by a cloud-based system, to an identity provider to authorize the user;

cause a display, via a unique user portal, of the one or more B2B applications that the user is authorized to access based on a response from the identity provider, wherein any applications of the one or more B2B applications that the user is unauthorized to access, based on the response from the identity provider, are omitted from the unique user portal, such that B2B applications that the user is unauthorized to access are invisible to the user; and

responsive to a selection of a B2B application of the one or more B2B applications and responsive to creation of a first tunnel from the B2B application to a broker in the cloud-based system, wherein the tunnel is an inside-out encrypted tunnel, stitch the first tunnel between the B2B application and the cloud-based system with a second tunnel between the user and the cloud-based system via an enforcement node, wherein the enforcement node has no visibility into traffic between the user and the B2B application, and wherein the B2B application and the user can only communicate with the cloud-based system, and application segmentation is achieved without network segmentation through user to application micro tunnels.

14. The cloud-based system of claim **13**, wherein real-time and historical activity and diagnostics of one or more users with the one or more B2B applications are logged and stored with activity and diagnostics from a plurality of users associated with the one or more B2B applications.

15. The cloud-based system of claim **13**, wherein the one or more B2B applications include business facing applications including any of Supply Chain Management (SCM) applications, inventory management applications, ordering applications, financial applications, and payroll applications.

16. The cloud-based system of claim **13**, wherein the one or more B2B applications are web-based applications and the request is a Uniform Resource Locator (URL).

17. The cloud-based system of claim **13**, wherein the user provides the request via a web browser executed on a user device.

18. The method of claim **1**, wherein creating the first tunnel includes creating an inside-out encrypted tunnel from an application connector associated with the B2B application to the cloud-based system, wherein the application connector dials out to the cloud-based system.

19. The method of claim **18**, further comprising

providing information to one or more nodes of the cloud-based system, a connector application executing on a user device associated with the user, and the application connector, wherein the instructions include certificates and details necessary to stitch secure connections; and

stitching the first tunnel between the application connector and the cloud-based system with a second tunnel between the connector application and the cloud-based system based on the information.

* * * * *